BAB III

ANALISA MASALAH DAN PERANCANGAN

III.1. Analisa

Sub bab ini berisikan tentang analisa sistem yang akan dibangun. Sub bab ini membahas teknik pemecahan masalah yang menguraikan sebuah sistem menjadi bagian-bagian komponen dengan tujuan mempelajari seberapa baik bagian-bagian komponen tersebut bekerja dan berinteraksi.

III.1.1. Analisa Masalah

Tujuan dari fase analisis adalah memahami dengan sebenar-benarnya kebutuhan dari sistem baru dan mengembangkan sebuah sistem yang mewadahi kebutuhan tersebut. Untuk mempermudah analisis sistem dalam menentukan kebutuhan secara lengkap, maka penulis membagi kebutuhan sistem kedalam dua jenis yakni, kebutuhan fungsional dan kebutuhan nonfungsional.

III.1.1.1. Analisa Kebutuhan Fungsional

Kebutuhan fungsional adalah jenis kebutuhan yang berisi proses-proses apa saja yang nantinya dilakukan oleh sistem. Kebutuhan fungsional juga berisi in*form*asi-in*form*asi apa saja yang harus ada dan dihasilkan oleh sistem. Berikut kebutuhan fungsional yang terdapat pada sistem yang dibangun :

1. Mengimplementasikan penggunaan *Visual Basic.Net 2008* dalam membuat perangkat lunak pengamanan *file* menggunakan algoritma AES.
2. Aplikasi harus dapat melakukan enkripsi terhadap sebuah *file *.xls* dan **doc*.

2. Aplikasi harus dapat melakukan dekripsi terhadap *file* yang sudah dienkripsi, tanpa merusak *file*.

III.1.1.2. Analisa Kebutuhan NonFungsional

Kebutuhan ini adalah tipe kebutuhan yang berisi properti perilaku yang dimiliki oleh sistem. Berikut adalah kebutuhan nonfungsional yang dimiliki sistem :

1. Operasional
   A. Dapat digunakan pada sistem operasi *Microsoft Windows XP/Vista/7* secara *stand alone*.
   B. Aplikasi dibangun dengan menggunakan komponen Paket *software Visual Basic 2008*
   C. Spesifikasi komputer *standard Processor Pentium IV* 2,6 GHz, Memori 512 MB, Kartu Grafik 128 MB
2. Kinerja

   Waktu yang diperlukan dalam mengeksekusi perangkat lunak pengamanan *file* menggunakan algoritma AES yang dibangun cukup ringan, sehingga eksekusi tampilannya cukup cepat.

III.2. Perancangan Sistem

Sub bab ini berisikan tentang rancangan sistem yang akan dibangun, dalam hal ini perancangan terhadap sistem.

III.2.1. Metode Algoritma AES

Algoritma kriptografi bernama Rijndael yang didesain oleh oleh Vincent Rijmen dan John Daemen asal Belgia keluar sebagai pemenang kontes algoritma kriptografi pengganti DES yang diadakan oleh NIST (National Institutes of Standards and Technology) milik pemerintah Amerika Serikat pada 26 November 2001. Algoritma Rijndael inilah yang kemudian dikenal dengan Advanced Encryption Standard (AES). Setelah mengalami beberapa proses standardisasi oleh NIST, Rijndael kemudian diadopsi menjadi standard algoritma kriptografi secara resmi pada 22 Mei 2002. Pada 2006, AES merupakan salah satu algoritma terpopuler yang digunakan dalam kriptografi kunci simetrik. AES ini merupakan algoritma block cipher dengan menggunakan sistem permutasi dan substitusi (P-Box dan S-Box) bukan dengan jaringan Feistel sebagaiman block cipher pada umumnya. Jenis AES terbagi 3, yaitu :

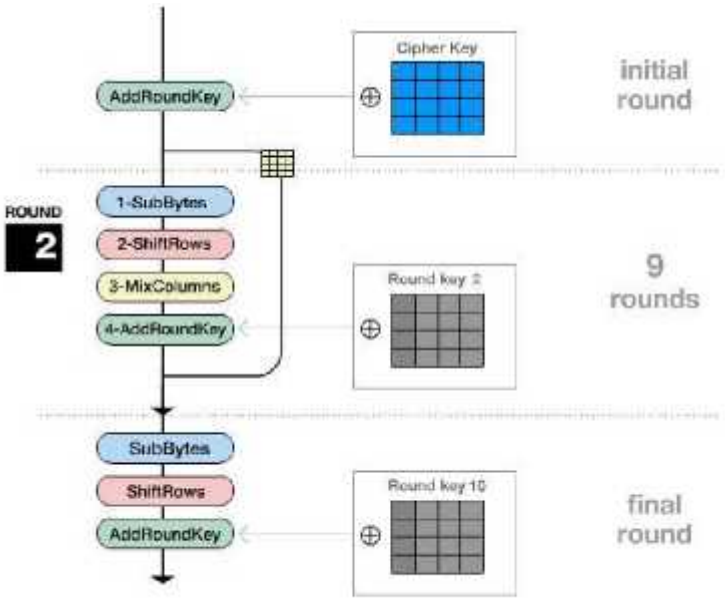
1.AES-128

2.AES-192

3.AES-256

Pengelompokkan jenis AES ini adalah berdasarkan panjang kunci yang digunakan. Angka-angka di belakang kata AES menggambarkan panjang kunci yang digunakan pada tipa-tiap AES. Selain itu, hal yang membedakan dari masing-masing AES ini adalah banyaknya round yang dipakai. AES-128 menggunakan 10 round, AES-192 sebanyak 12 round, dan AES-256 sebanyak

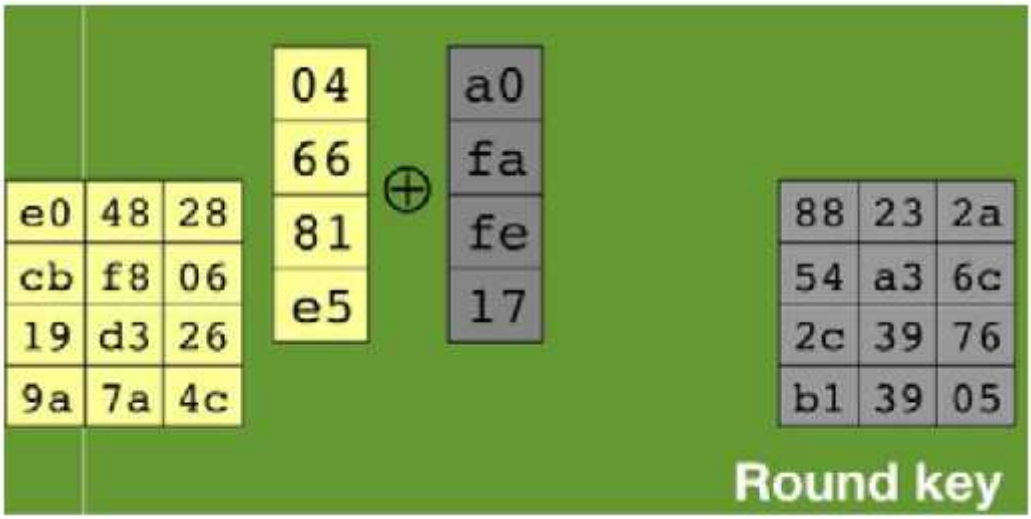
14 round. AES memiliki ukuran block yang tetap sepanjang 128 bit dan ukuran kunci sepanjang 128, 192, atau 256 bit. Tidak seperti Rijndael yang block dan kuncinya dapat berukuran kelipatan 32 bit dengan ukuran minimum 128 bit dan maksimum 256 bit. Berdasarkan ukuran block yang tetap, AES bekerja pada matriks berukuran 4x4 di mana tiap-tiap sel matriks terdiri atas 1 byte (8 bit). Sedangkan Rijndael sendiri dapat mempunyai ukuran matriks yang lebih dari itu dengan menambahkan kolom sebanyak yang diperlukan. Blok chiper tersebut dalam pembahasan ini akan diasumsikan sebagai sebuah kotak. Setiap plainteks akan dikonversikan terlebih dahulu ke dalam blok-blok tersebut dalam bentuk heksadesimal. Barulah kemudian blok itu akan diproses dengan metode yang akan dijelaskan. Secara umum metode yang digunakan dalam pemrosesan enkripsi dalam algoritma ini dapat dilihat melalui Gambar berikut :



Gambar III. 1. *Diagram* AES

III.2.2 ADD ROUND KEY

Add Round Key pada dasarnya adalah mengkombinasikan chiper teks yang sudah ada Dengan chiper key yang chiper key dengan hubungan XOR. Bagannya bisa dilihat pada gambar di bawah ini :



Gambar III.2. ***Round Key***

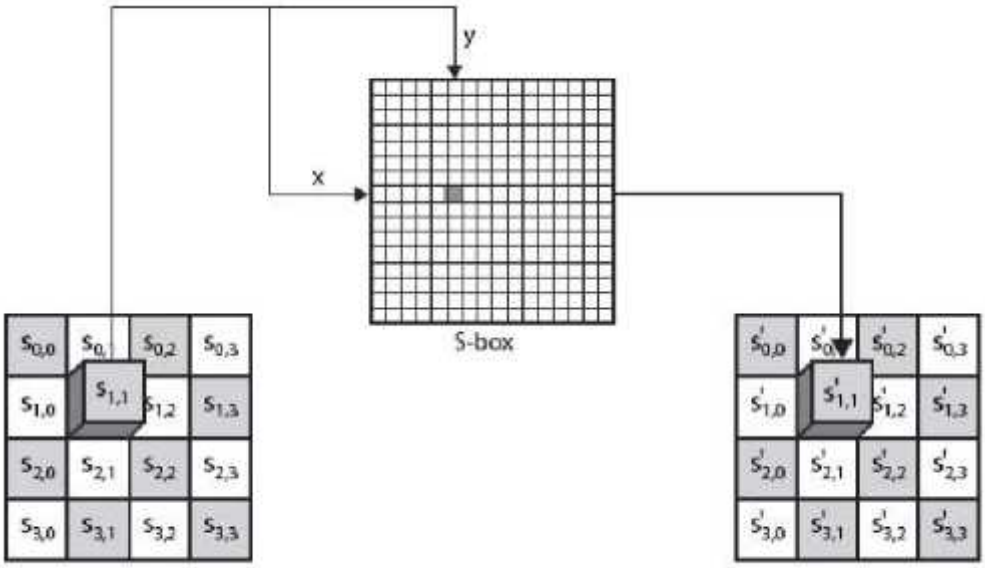
Pada gambar tersebut di sebelah kiri adalah chiper teks dan sebelah kanan adalah round key nya. XOR dilakukan per kolom yaitu kolom-1 chiper teks di XOR dengan kolom-1 round key dan seterusnya.

III.2.3 SUB BYTES

Prinsip dari Sub Bytes adalah menukar isi matriks/tabel yang ada dengan matriks/tabel lain yang disebut dengan Rijndael S-Box. Di bawah ini adalah contoh Sub Bytes dan Rijndael S-Box.

	x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	xa	xb	xc	xd	xe	xf
0x	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
1x	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
2x	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
3x	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
4x	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
5x	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
6x	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
7x	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
8x	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
9x	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
ax	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
bx	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
cx	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
dx	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
ex	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
fx	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

Gambar III.3. ***Rijndael S-Box***



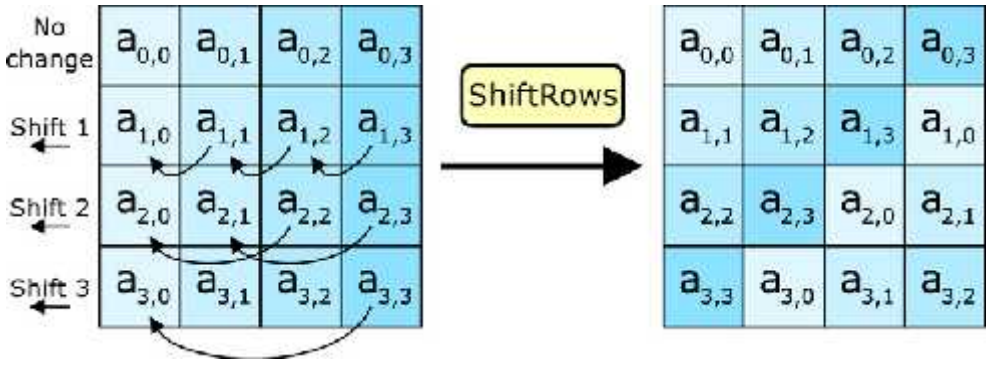
Gambar III.4 . ***Ilustrasi Sub Bytes***

Gambar III.3 adalah contoh dari Rijndael S-Box, di sana terdapat nomor kolom dan nomor baris. Seperti yang telah disebutkan sebelumnya, tiap isi kotak dari blok chiper berisi informasi dalam bentuk heksadesimal yang terdiri dari dua digit, bisa angka-angka, angka-huruf, ataupun huruf-angka yang semuanya tercantum dalam Rijndael S-Box. Langkahnya adalah mengambil salah satu isi kotak matriks, mencocokkannya dengan digit kiri

sebagai baris dan digit kanan sebagai kolom. Kemudian dengan mengetahui kolom dan baris, kita dapat mengambil sebuah isi tabel dari Rijndael S-Box. Langkah terakhir adalah mengubah keseluruhan blok chiper menjadi blok yang baru yang isinya adalah hasil penukaran semua isi blok dengan isi langkah yang disebutkan sebelumnya.

III.2.4 SHIFT ROWS

Shift Rows seperti namanya adalah sebuah proses yang melakukan shift atau pergeseran pada setiap elemen blok/tabel yang dilakukan per barisnya. Yaitu baris pertama tidak dilakukan pergeseran, baris kedua dilakukan pergeseran 1 byte, baris ketiga dilakukan pergeseran 2 byte, dan baris keempat dilakukan pergeseran 3 byte. Pergeseran tersebut terlihat dalam sebuah blok adalah sebuah pergeseran tiap elemen ke kiri tergantung berapa byte tergesernya, tiap pergeseran 1 byte berarti bergeser ke kiri sebanyak satu kali. Ilustrasi dari Tahap ini diperlihatkan oleh gambar di bawah ini.



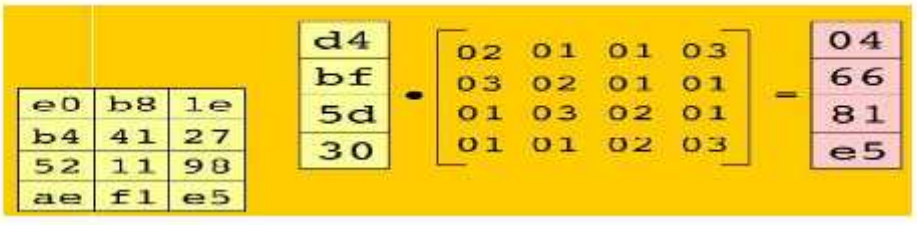
Gambar III.5. Ilustrasi dari *Shift Row*

III.2.5 MIX COLUMNS

Yang terjadi saat Mix Column adalah mengalikan tiap elemen dari blok chiper dengan matriks yang ditunjukkan oleh Gambar III.6 Tabel sudah ditentukan dan siap pakai. Pengalian dilakukan seperti perkalian matriks biasa yaitu menggunakan dot product lalu perkalian keduanya dimasukkan ke dalam sebuah blok chiper baru. Ilustrasi dalam gambar 12 akan menjelaskan mengenai bagaimana perkalian ini seharusnya dilakukan. Dengan begitu seluruh rangkaian proses yang terjadi pada AES telah dijelaskan dan selanjutnya adalah menerangkan mengenai penggunaan tiap-tiap proses tersebut.

02	01	01	03
03	02	01	01
01	03	02	01
01	01	02	03

Gambar III.6. Tabel untuk *Mix Columns*

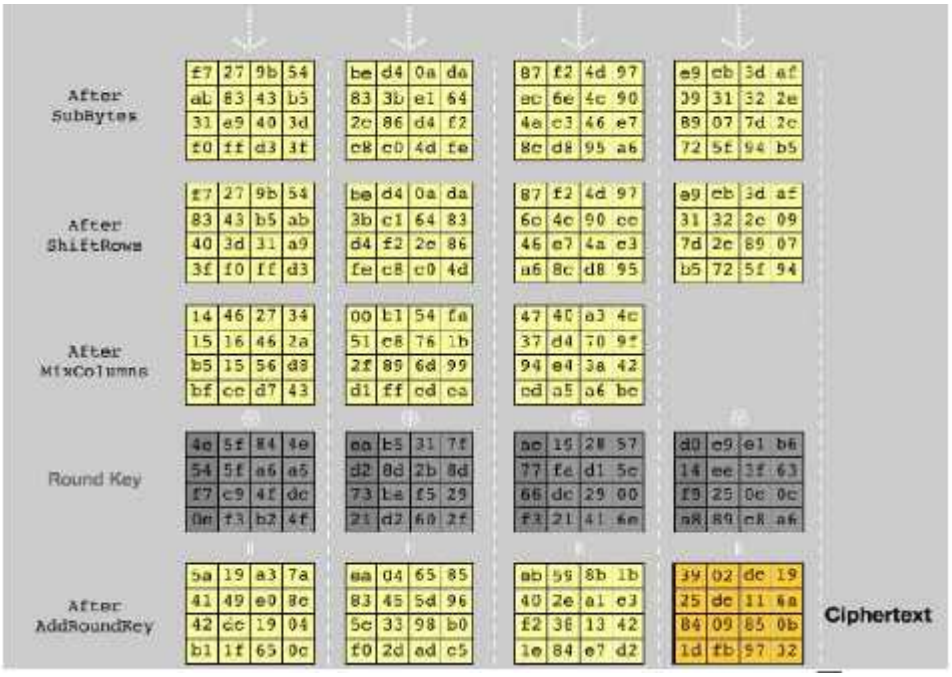


Gambar III.7. Ilustrasi *Mix Columns*

III.2.6 DIAGRAM ALIR AES

Kembali melihat diagram yang ditunjukkan oleh Gambar III.8 Seperti yang terlihat semua proses yang telah dijelaskan sebelumnya terdapat pada diagram tersebut. Yang artinya adalah mulai dari ronde kedua, dilakukan pengulangan terus menerus dengan rangkaian proses Sub Bytes, Shift Rows, Mix Columns, dan Add Round Key, setelah itu hasil dari ronde tersebut akan digunakan pada ronde berikutnya dengan metode yang sama. Namun pada ronde kesepuluh, Proses Mix Columns tidak dilakukan, dengan kata lain urutan proses yang dilakukan adalah Sub Bytes, Shift Rows, dan Add Round Key, hasil dari Add Round Key inilah yang dijadikan sebagai chiperteks dari AES. Lebih jelasnya bisa dilihat dengan Gambar 13 dan 14 yang akan menerangkan mengenai kasus tersebut.

Gambar III.8. Ilustrasi Ronde 2 hingga Ronde 6

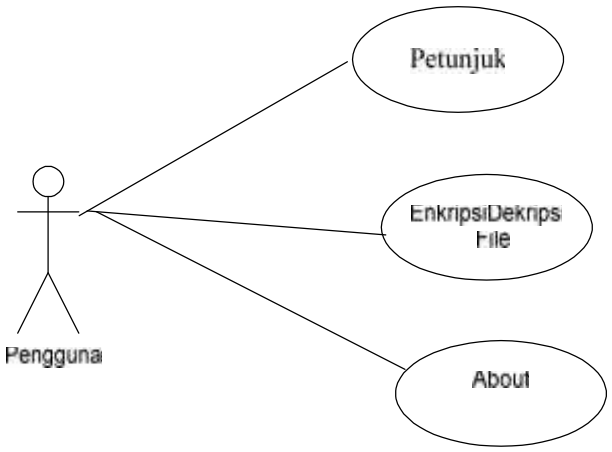


Gambar III.9. Ilustrasi Ronde 7 hingga Ronde 10

Dengan mengetahui semua proses yang ada pada AES, maka kita dapat menggunakannya dalam berbagai contoh kasus yang muncul di kehidupan sehari-hari.

III.3.1. Diagram *Use Case*

Berikut ini merupakan diagram *use case* dari "*Perangkat Lunak Pengamanan File Menggunakan Algoritma AES*". Terlihat pada gambar III.10.

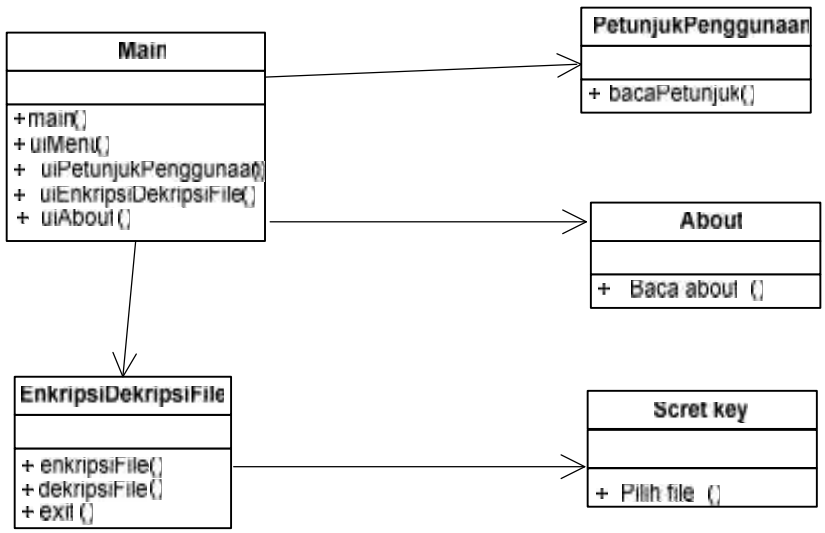


Gambar III.10. Diagram Aplikasi

Pada diagram *use case* di atas, aktor yang didefenisikan pada aplikasi hanya 1 yakni pengguna. pengguna adalah orang yang menjalankan aplikasi. Ketika aplikasi dijalankan, aplikasi akan menampilkan halaman dan mengeksekusi perintah sesuai dengan *event* yang diberikan *pengguna* pada *interface* aplikasi.

III.3.2. Diagram Kelas

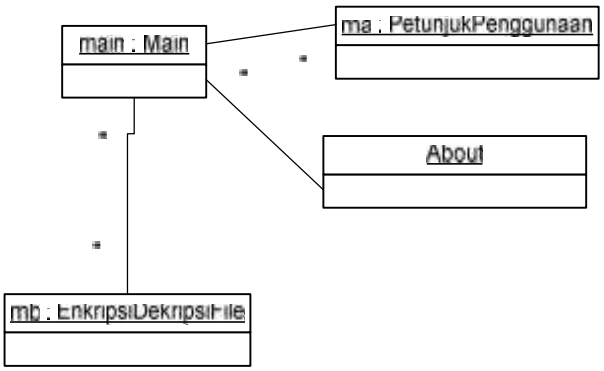
Berikut ini merupakan diagram kelas dari aplikasi "*Perangkat Lunak Pengamanan File Menggunakan Algoritma AES*". Terlihat pada gambar III.11.



Gambar III.11. Diagram Kelas Aplikasi

III.3.3. Diagram Objek

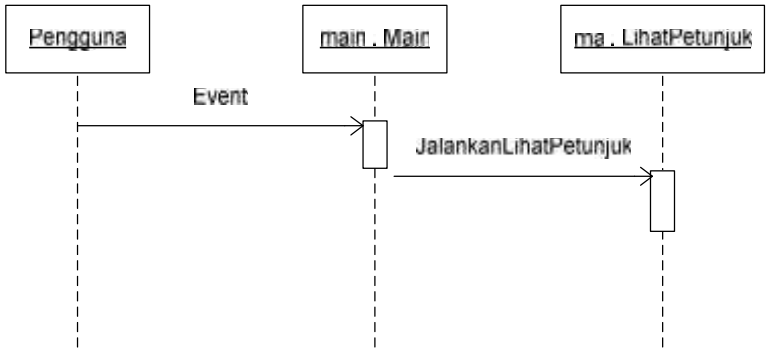
Berikut ini merupakan diagram objek dari aplikasi "*Perangkat Lunak Pengamanan File Menggunakan Algoritma AES*". Terlihat pada gambar III.12.



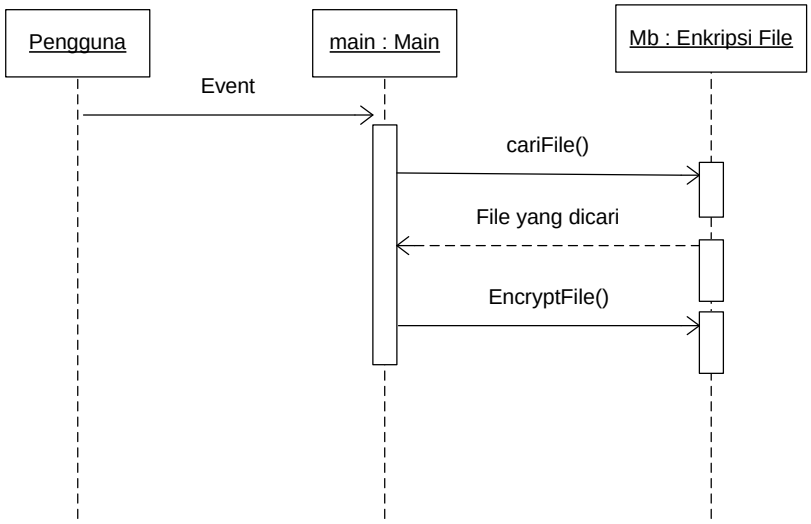
Gambar III.12. Diagram Objek Aplikasi

III.3.4. Diagram Sekuen

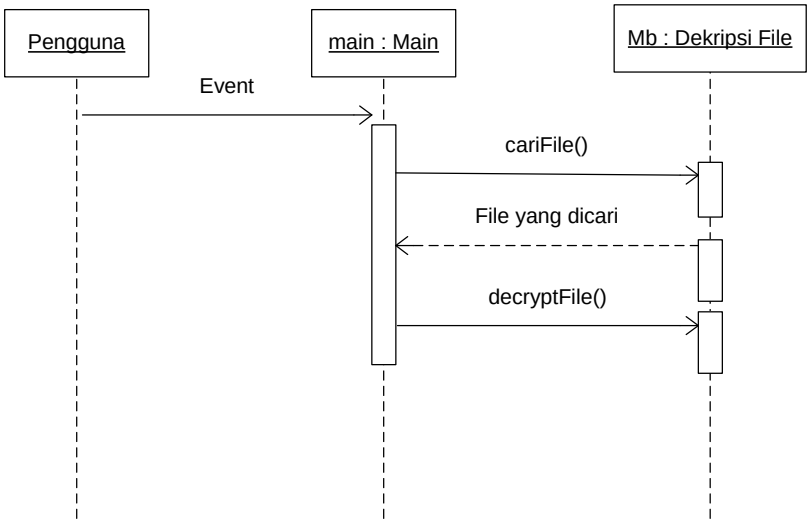
Berikut ini merupakan diagram *sequence* dari aplikasi "*Perangkat Lunak Pengamanan File Menggunakan Algoritma AES*". Terlihat pada gambar III.13. dan gambar III.14.



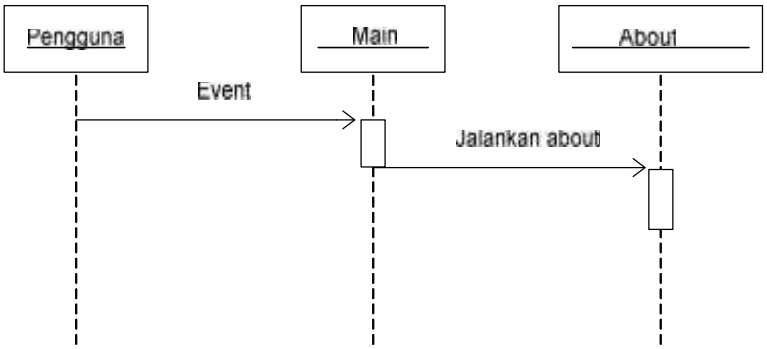
Gambar III.13. *Diagram Sequence* Untuk Petunjuk Penggunaan



Gambar III.14. Diagram *Sequence* Untuk Enkripsi *File*



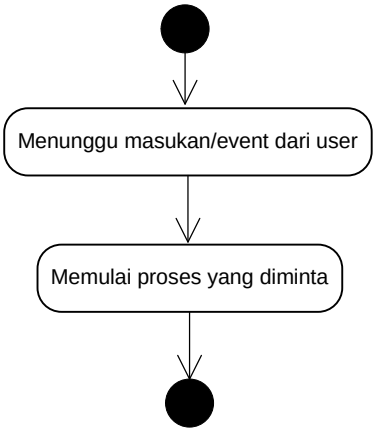
Gambar III.15. Diagram *Sequence* Untuk Dekripsi *File*



Gambar III.16. Diagram Sequence Untuk About

III.3.5. Diagram Status

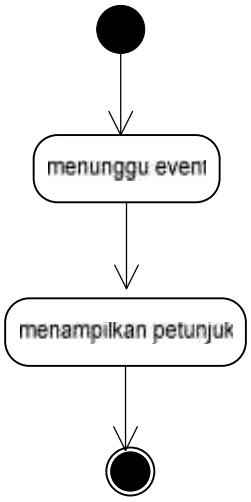
Berikut ini merupakan diagram status dari aplikasi "*Perangkat Lunak Pengamanan File Menggunakan Algoritma AES*".



Gambar III.17. Diagram Status Untuk Objek : main dari kelas Main

Keterangan :

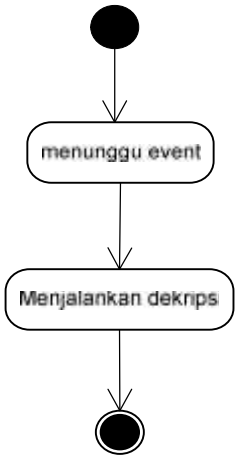
Ui merupakan inisialisasi untuk objek aplikasi. Ui merupakan singkatan dari pengguna interface.



Gambar III.18. Diagram Status Untuk Objek : ma dari kelas Petunjuk Penggunaan

Gambar III.19. Diagram Status Untuk Objek : mc dari kelas *About*

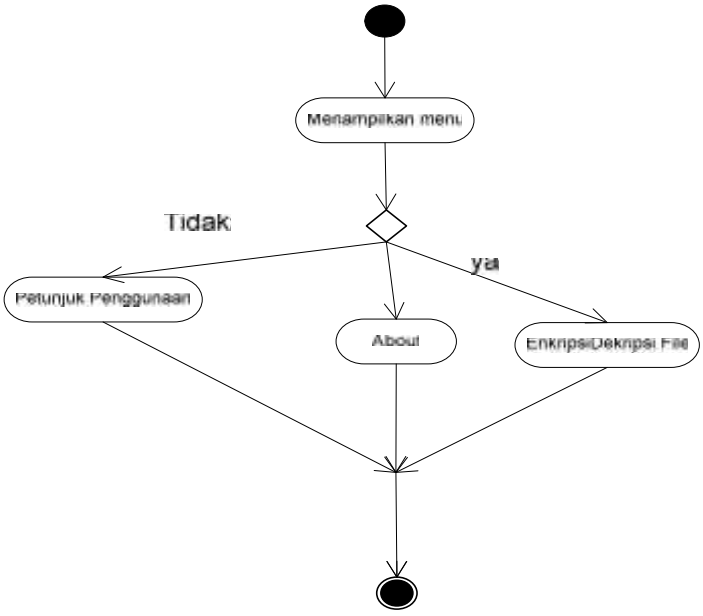
Gambar III.20. Diagram Status Untuk Objek : mb dari kelas Enkripsi *File*



Gambar III.21. Diagram Status Untuk Objek : mb dari kelas Dekripsi *File*

III.3.6. Diagram Aktivitas

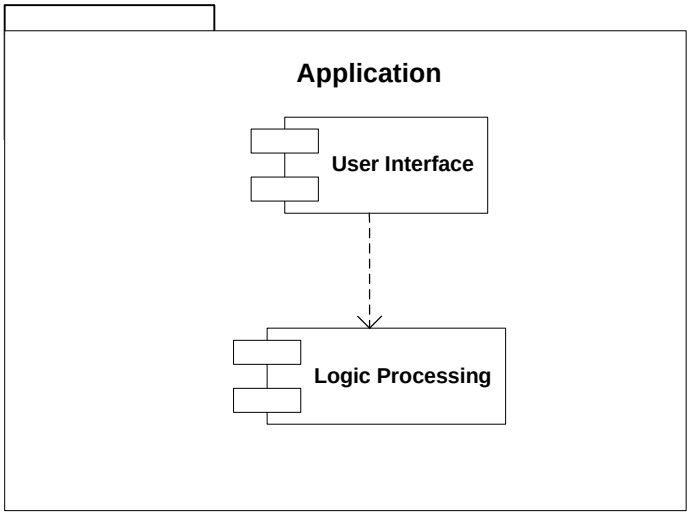
Berikut ini merupakan diagram aktivitas dari aplikasi "*Perangkat Lunak Pengamanan File Menggunakan Algoritma AES*". Terlihat pada gambar III.13



Gambar III.22. Diagram Aktivitas

III.3.7. Diagram Komponen

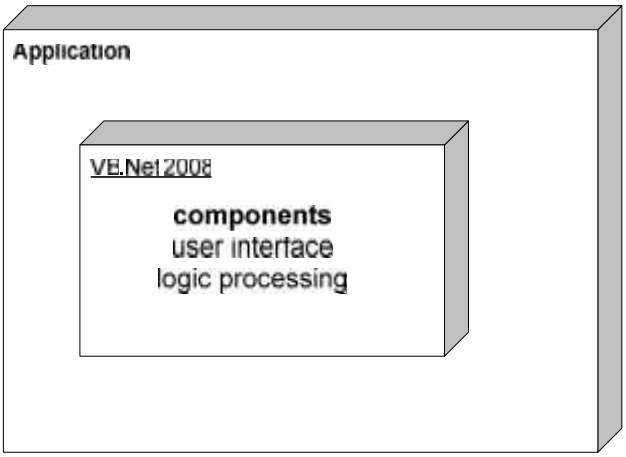
Berikut ini merupakan diagram komponen dari aplikasi "*Perangkat Lunak Pengamanan File Menggunakan Algoritma AES*". Terlihat pada gambar III.14



Gambar III.23. Diagram Komponen

III.3.8. *Diagram Deployment*

Berikut ini merupakan diagram *deployment* dari aplikasi "*Perangkat Lunak Pengamanan File Menggunakan Algoritma AES*". Terlihat pada gambar III.15



Gambar III.24. *Diagram Deployment*

III.4 Perancangan Tampilan

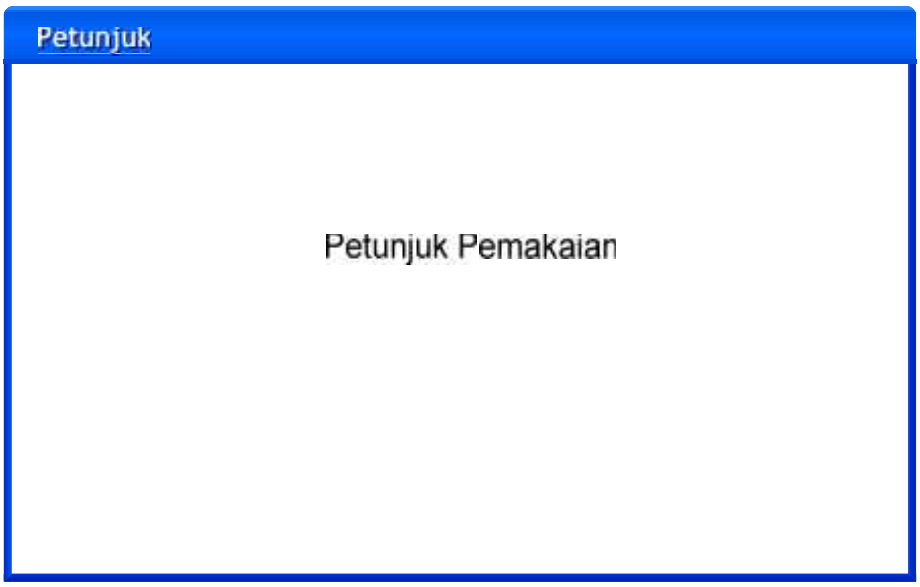
III.4.1 Rancangan Tampilan *Form Menu* Pilihan

Rancangan *Form menu* pilihan berisi *menu-menu* pilihan yang ada pada aplikasi, *menu-menu* yang terdapat pada *form menu* pilihan adalah *menu* petunjuk, *enkripsi-deskripsi file*, dan *about* jika masing-masing *menu* dipilih maka *form* yang berkaitan dengan *menu* yang dipilih akan muncul serta *menu exit* untuk keluar dari aplikasi. Terlihat pada gambar III.25.

Gambar III.25. Rancangan *Form Menu* Pilihan

III.4.2 Rancangan Tampilan *Form* Petunjuk

Form petunjuk merupakan *form* yang berisi petunjuk penggunaan aplikasi sehingga bila ada pengguna baru akan membuka aplikasi pengguna tersebut tidak perlu bertanya kepada pengguna sebelumnya. Terlihat pada gambar III.26.



Gambar III.26. Rancangan *Form Menu* Petunjuk

III.4.3. Rancangan Tampilan *Form Key*

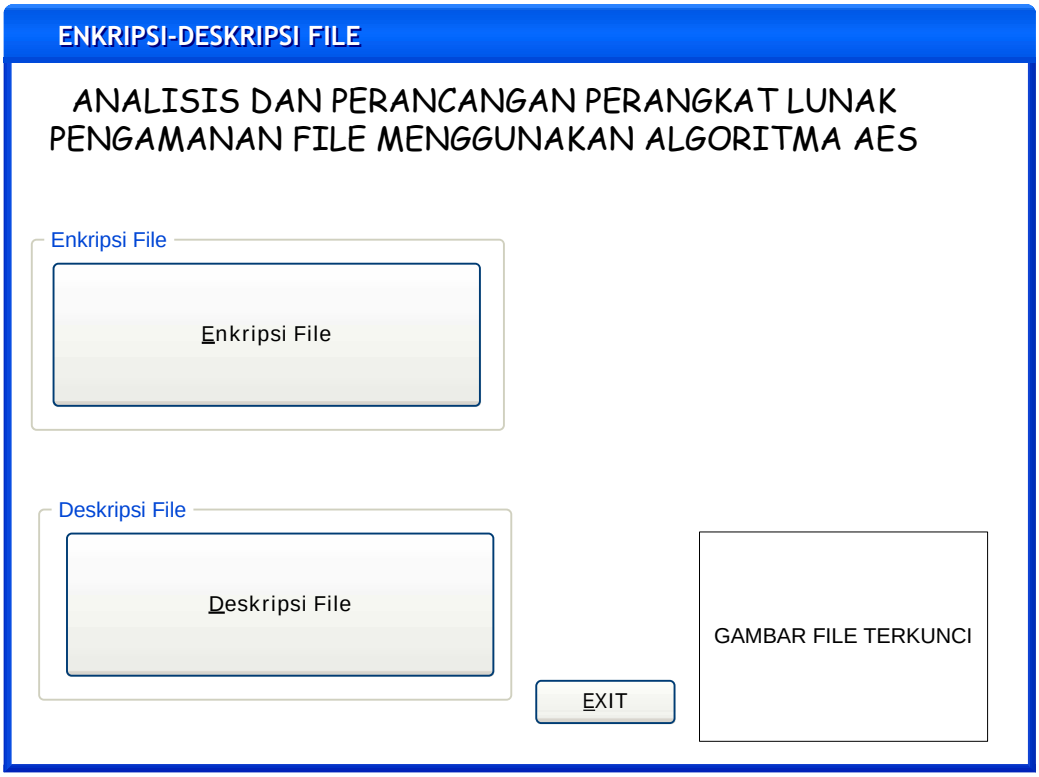
Form key merupakan *form* pembatas akses terhadap aplikasi jika pengguna ingin melakukan enkripsi atau deskripsi *file* maka pengguna harus mengisi teks *key* dengan kata kunci yang benar.

Gambar III.27. Rancangan *Form Menu* Petunjuk

III.4.4. Rancangan Tampilan *Form* Utama

Form utama merupakan *form* yang berfungsi untuk melakukan proses enkripsi dan deskripsi *file*. Jika pengguna ingin malakukan proses enkripsi maka pengguna dapat menekan tombol “Enkripsi *File*” kemudian akan muncul jendela *Explorer* selanjutnya pilih *file* yang akan dienkripsi kemudian tekan tombol

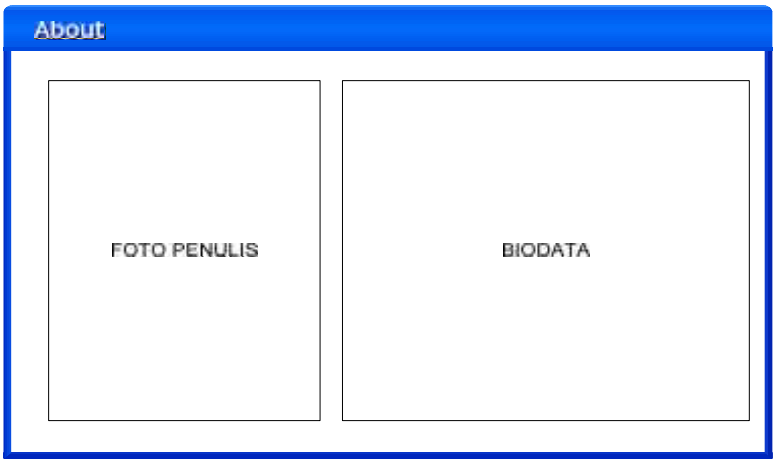
"*Open*" maka secara otmomatis *file* tersebut akan terenkripsi dan berubah *extention*. Namun jika pengguna ingin malakukan proses deskripsi terhadap *file* yang telah terenkrip maka pengguna dapat menekan tombol "Deskripsi *File*" kemudian akan muncul jendela *Explorer* selanjutnya pilih *file* yang akan dideskripsi kemudian tekan tombol "*Open*" maka secara otmomatis *file* tersebut akan terdeskripsi dan *extention file* tersebut kembali seperti semula. Jika pengguna ingin keluar dari aplikasi ini maka pengguna dapat menekan tombol "*EXIT*". Terlihat pada gambar III.28.



Gambar III.28. Perancangan *Form* Utama

III.4.5. Rancangan Tampilan *Form About*

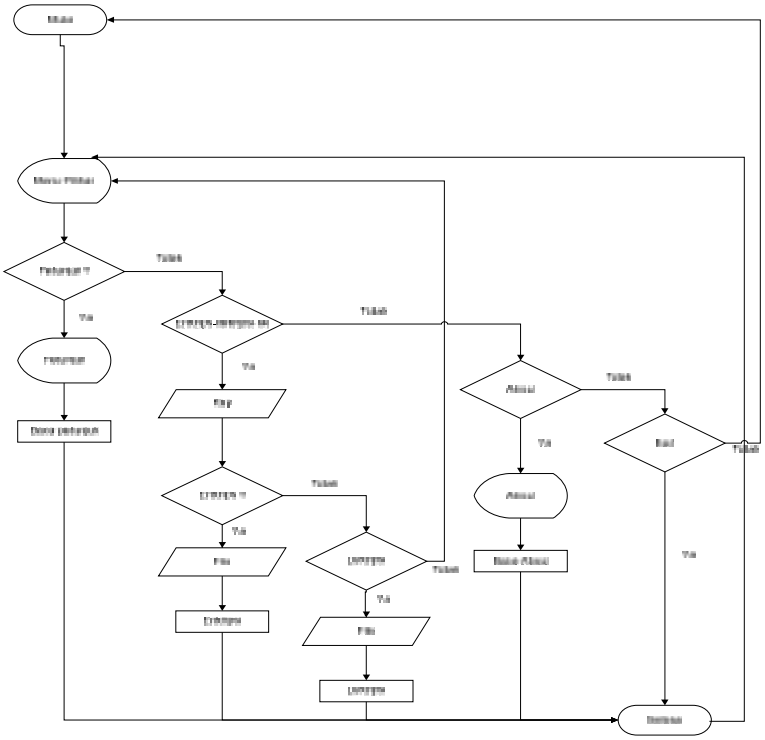
Form about merupakan *form* yang berisi tentang biodata penulis/*programmer* aplikasi.



Gambar III.29. Perancangan *Form About*

III.5 Perancangan *FlowChart*

Berikut ini merupakan *flowchart* aplikasi, Terlihat pada gambar III.30.



Gambar III.30. *Flowchart* Aplikasi Analisis Dan Perancangan Perangkat Lunak Pengamanan *File* Menggunakan Algoritma AES