BAB III
ANALISIS DAN PERANCANGAN

III.1. Analisis Masalah

Proses enkripsi di sini diartikan sebagai proses perubahan dari suatu pesan asli (*plain text)* menjadi suatu pesan yang terlindungi dalam hal ini pesan yang tersandi (*chiper text),* sedangkan untuk proses dekripsi adalah suatu proses pengembalian pesan tersandi yang terlindungi menjadi bentuk data asli pesan tersebut. Berdasarkan situasi dan kondisi yang tercermin pada latar belakang, menjelaskan sebuah pokok permasalahan yaitu celah keamanan pada file gambar sangat besar yang menyebabkan duplikasi data yang tidak dapat diketahui tingkat keamanan data tersebut.

Penelitian ini difokuskan pada perancangan sistem untuk mengamankan gambar dengan menggunakan algoritma *AES (Advanced Encryption Standart)* untuk mengenkripsi dan mendekirpsi gambar. Sistem dibangun berbasis *web* yang bertujuan untuk membantu pengguna agar mudah mengakses aplikasi yang dibuat karena bersifat *online* dan *multiuser*. Dengan demikian, hasil penelitian ini dapat memberikan kontribusi kepada publik dalam hal sarana pengamanan gambar, yaitu untuk melindungi gambar yang bersifat pribadi dan tidak seharusnya menjadi konsumsi masyarakat luas serta kemudahan akses aplikasi.

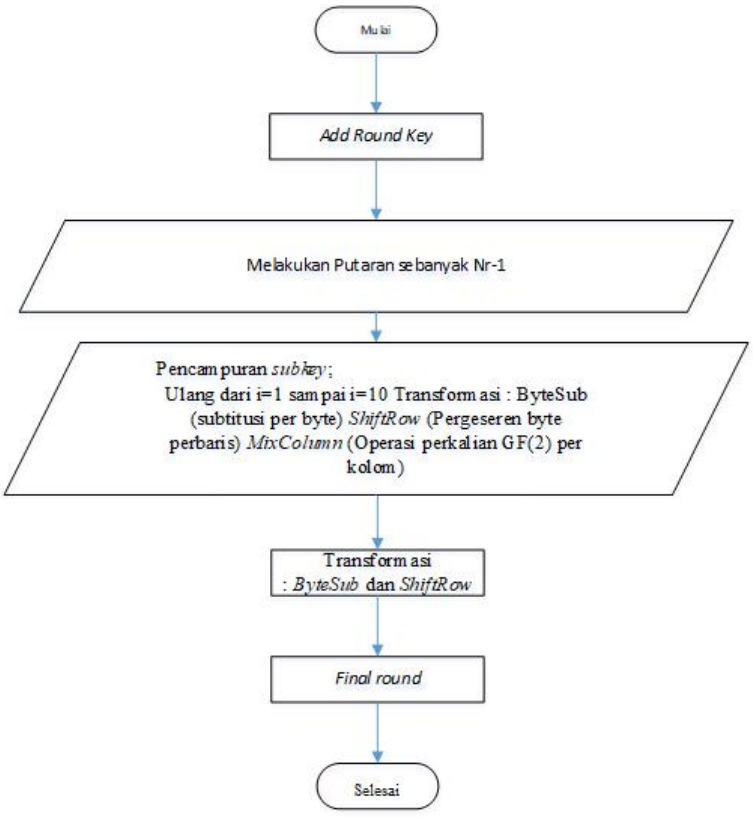
III.1.1. Strategi Pemecahan Masalah

Strategi dalam melakukan pemecahan masalah yang sedang dianalisa oleh penulis mengenai perancangan Aplikasi Pengamanan *Image* Menggunakan Metode Algoritma AES (*Advanced Encryption Standard*) Berbasis Web adalah sebagai berikut :

1. Merancang sebuah aplikasi dengan memanfaatkan sistem keamanan data yang dapat menjaga kerahasiaan dan keamanan data.
2. Merancang dan membangun sebuah aplikasi keamanan gambar dengan menggunakan Algoritma AES.

III.2. Penerapan Algoritma AES

III.2.1. Flowchart Algoritma



Gambar III. 1. *Flowchart AES*

III.2.2. Studi Kasus Algoritma *AES*

Untuk mendapatkan *plaintext* pada gambar berformat jpg, piksel pada gambar akan dirubah kedalam bentuk matrik 4x4. Untuk proses yang lebih cepat mencari parameter pada citra data masukan, citra masukan awal akan diubah ukurannya dengan proses resizing dari ukuran citra asli menjadi ukuran 4x4 piksel dengan mode warna RGB (Red, Green, Blue) yang artinya dalam setiap piksel terdapat 3 nilai warna.

Gambar III. 2. Gambar Enkripsi

Rumus yang digunakan untuk proses merubah gambar menjadi matrik sebagai berikut :

X = 0.21*R + 0.72*G + 0.07*B

Adapun langkah-langkahnya dapat diilustrasikan sebagai berikut:

1. Nilai RGB pada piksel gambar dimasukkan kedalam sebuah matrik dengan ordo 4x4.

R=,215, G=,15, B=,52	R=,219, G=,27, B=,66	R=,228, G=,42, B=,81	R=,235, G=,53, B=,94
R=,220, G=,20, B=,57	R=,233, G=,41, B=,80	R=,251, G=,65, B=,104	R=,255, G=,81, B=,122
R=,206, G=,41, B=,58	R=,234, G=,89, B=,104	R=,248, G=,117, B=,131	R=,253, G=,126, B=,143
R=,220, G=,55, B=,72	R=,254, G=,109, B=,124	R=,255, G=137, B=,151	R=,255, G=,140, B=,157

Gambar III. 3. Susunan RGB pixel gambar

Perhitungan nilai matrik :

Perhitungan piksel (0.0) :

X = 0,21 * 215 + 0,72 * 15 + 0,07 * 52

X = 45,15 + 10,8 + 3,64

X = 59,59 = 60

Perhitungan piksel (0.1) :

X = 0,21 * 219 + 0,72 * 27 + 0,07 * 66

X = 45,99 + 19,44 + 4,62

X = 70,05 = 70

Perhitungan piksel (0.2) :

X = 0,21 * 228 + 0,72 * 42 + 0,07 * 81

X = 47,88 + 30,24 + 6,58

X = 84,7 = 85

Perhitungan piksel (0.3) :

X = 0,21 * 235 + 0,72 * 53 + 0,07 * 94

X = 49,35 + 38,16 + 6,58

X = 94,09 = 94

Perhitungan piksel (1.0) :

X = 0,21 * 220 + 0,72 * 20 + 0,07 * 57

X = 46,2 + 14,4 + 3,99

X = 64,59 = 65

Perhitungan piksel (1.1) :

X = 0,21 * 233 + 0,72 * 41 + 0,07 * 80

X = 48,93 + 29,52 + 5,6

X = 84,05 = 84

Perhitungan piksel (1.2) :

X = 0,21 * 251 + 0,72 * 65 + 0,07 * 104

X = 52,71 + 46,8 + 7,28

X = 106,79 = 107

Perhitungan piksel (1.3) :

X = 0,21 * 255 + 0,72 * 81 + 0,07 * 122

X = 53,55 + 58,32 + 8,54

X = 120,41 = 120

Perhitungan piksel (2.0) :

X = 0,21 * 206 + 0,72 * 41 + 0,07 * 58

X = 43,26 + 29,52 + 4,06

X = 76,84 = 77

Perhitungan piksel (2.1) :

X = 0,21 * 234 + 0,72 * 89 + 0,07 * 104

X = 49,14 + 29,52 + 7,28

X = 85, 94 = 86

Perhitungan piksel (2.2) :

X = 0,21 * 248 + 0,72 * 117 + 0,07 * 131

X = 52,08 + 84,24 + 9,17

X = 145,49 = 145

Perhitungan piksel (2.3) :

X = 0,21 * 253 + 0,72 * 126 + 0,07 * 143

X = 53,13 + 90,72 + 10,01

X = 153,86 = 154

Perhitungan piksel (3.0) :

X = 0,21 * 220 + 0,72 * 55 + 0,07 * 72

X = 46,2 + 39,6 + 5,04

X = 90,84 = 91

Perhitungan piksel (3.1) :

X = 0,21 * 254 + 0,72 * 109 + 0,07 * 124

X = 53,34 + 78,48+ 8,68

X = 140,5 = 141

Perhitungan piksel (3.2) :

X = 0,21 * 255 + 0,72 * 137 + 0,07 * 151

X = 53, 55 + 98,64 + 10,57

X = 162, 76 = 163

Perhitungan piksel (3.3) :

X = 0,21 * 255 + 0,72 * 140 + 0,07 * 157

X = 53,55 + 100,8 + 10,99

X = 165,34 = 165

60	70	85	94
65	84	107	120
77	86	145	154
91	141	163	165

2. Setelah mendapatkan nilai matrik pada gambar. Maka nilai matrik tersebut akan dijadikan *plaintext.*

60	70	85	94
65	84	107	120
77	86	145	154
91	141	163	165

Setelah dilakukan percobaan terhadap sebuah file gambar berformat jpg untuk proses enkripsi maka didapatkan hasil sebagai berikut. Contoh dilakukan percobaan untuk proses enkripsi terhadap file airplane.jpg dengan resolusi 4 x 4 pixel dan size 9,07 kb

PlainText : 60 65 77 91 | 70 84 86 141 | 85 107 145 163 | 94 120 154 165

Key : 2b 7e 15 16 | 28 ae d2 a6 | - - - - | - - - -

Tabel III. 1. Tabel RCon

RCON [1]	RCON [2]	RCON [3]	RCON [4]	RCON [5]	RCON [6]	RCON [7]	RCON [8]	RCON [9]	RCON [10]
[01]	[02]	[04]	[08]	[10]	[20]	[40]	[80]	[1b]	[36]
[00]	[00]	[00]	[00]	[00]	[00]	[00]	[00]	[00]	[00]
[00]	[00]	[00]	[00]	[00]	[00]	[00]	[00]	[00]	[00]
[00]	[00]	[00]	[00]	[00]	[00]	[00]	[00]	[00]	[00]

Tabel III. 2. Tabel S-Box

S-Box Values																
S(xy)		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

Pertama hitung dulu key schedule. Hasil dari perhitungan akan digunakan pada proses selanjutnya untuk peoses enkripsi.

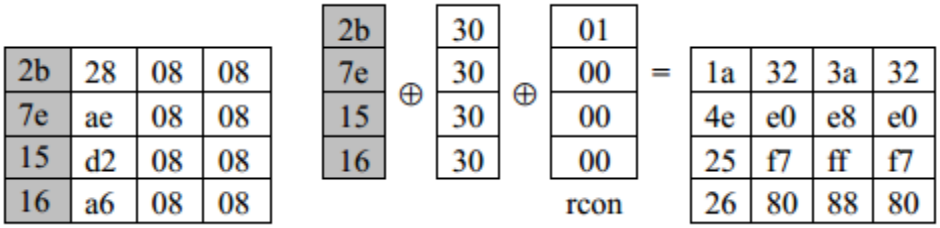
Key Schedule

Cipher Key : 2b 7e 15 16 | 28 ae d2 a6 | 08 08 08 08 | 08 08 08 08

2b	28	08	08
7e	ae	08	08
15	d2	08	08
16	a6	08	08

RotWord : 08 08 08 08

S-Box : 30 30 30 30



Round Key 1 : 1a 4e 25 26 | 32 e0 f7 80 | 3a e8 ff 88 | 32 e0 f7 80

Round Key 2 : f9 26 e8 05 | cb c6 1f 85 | f1 2e e0 0d | c3 ce 17 8d

Round Key 3 : 76 d6 b5 2b | bd 10 aa ae | 4c 3e 4a a3 | 8f f0 5d 2e

Round Key 4 : f2 9a 84 58 | 4f 8a 2e f6 | 03 b4 64 55 | 8c 44 39 7b

Round Key 5 : f9 88 a5 3c | b6 02 8b ca | b5 b6 ef 9f | 39 f2 d6 e4

Round Key 6 : 50 7e cc 2e | e6 7c 47 e4 | 53 ca a8 7b | 6a 38 7e 9f

Round Key 7 : 17 8d 17 2c | f1 f1 50 c8 | a2 3b f8 b3 | c8 03 86 2c

Round Key 8 : ec c9 66 c4 | 1d 38 36 0c | bf 03 ce bf | 77 00 48 93

Initial Round

Pertama dilakukan proses inisialisasi dengan operasi XOR antara *State* dan *Key*. Jika *state* atau *key* kurang dari 16 *byte* maka isi *byte* yang kosong dengan aturan *padding* PKCS5.

Initial State : 32 43 f6 a8 | 88 5a 30 8d | 31 31 98 a2 | 04 04 04 04

Cipher Key : 2b 7e 15 16 | 28 ae d2 a6 | 08 08 08 08 | 08 08 08 08

After AddRoundKey : 19 3d e3 be | a0 f4 e2 2b | 39 39 90 aa | 0c 0c 0c 0c

Round 1

State : 19 3d e3 be | a0 f4 e2 2b | 39 39 90 aa | 0c 0c 0c 0c

Round Key : 1a 4e 25 26 | 32 e0 f7 80 | 3a e8 ff 88 | 32 e0 f7 80

Setelah tahap inisialisasi maka dimulai round 1, tahap pertama dalam setiap round atau putaran adalah SubBytes, yaitu subtitusi State menggunakan SBox. Seperti yang terlihat pada tabel berikut ini , Cara mencarinya sebagai berikut:

State Round 1 memiliki matriks { 19 3d e3 be | a0 f4 e2 2b | 39 39 90 aa | 0c 0c 0c 0c }, jika diketahui byte pertama 19 y = 1 , x = 9, maka :

	x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	xa	xb	xc	xd	xe	xf
0y	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

Begitu seterusnya sampai semua matriks tersubtitusi,

After SubBytes : d4 27 11 ae | e0 bf 98 f1 | 12 12 60 ac | fe fe fe fe

Tahap selanjutnya adalah *ShiftRows*,

d4	e0	12	fe	→	d4	e0	12	fe	→	d4	e0	12	fe
27	bf	12	fe		bf	12	fe	27		bf	12	fe	27
11	98	60	fe		**11**	**98**	60	fe		60	fe	11	98
ae	f1	ac	fe		ae	f1	ac	fe		**ae**	**f1**	**ac**	fe

→	d4	e0	12	fe
	bf	12	fe	27
	60	fe	11	98
	fe	**ae**	**f1**	**ac**

After ShiftRows : d4 bf 60 fe | e0 12 fe ae | 12 fe 11 f1 | fe 27 98 ac

Selanjutnya adalah tahap MixColumns yaitu perkalian matriks antara hasil ShiftRows dan GF(8) atau matriks gaulios field yang telah ditentukan,

					Matriks GF(8)			
d4	e0	12	fe		02	01	01	03
bf	12	fe	27	x	03	02	01	01
60	fe	11	98		01	03	02	01
fe	ae	f1	ac		01	01	02	03

After MixColumns : f7 ef b2 5f | bd 73 fc 90 | dd 37 c6 20 | ba af 1d e5

After AddRoundKey : ed a1 97 79 | 8f 93 0b 10 | e7 df 39 a8 | 88 4f ea 65

c. Round 2

After SubBytes : 55 32 88 b6 | 73 dc 2b ca | 94 9e 12 c2 | c4 84 87 4d

After ShiftRows : 55 dc 12 4d | 73 9e 87 b6 | 94 84 88 ca | c4 32 2b c2

After MixColumns : 8a 8d 7a ab | 6e 70 39 fb | e6 ce 5e 24 | 2c 1f fd d1

After AddRoundKey : 73 ab 92 ae | a5 b6 26 7e | 17 e0 be 29 | ef d1 ea 5c

d. Round 3

After SubBytes : 8f 62 4f e4 | 06 4e f7 f3 | f0 e1 ae a5 | df 3e 87 4a

After ShiftRows : 8f 4e ae 4a | 06 e1 87 e4 | f0 3e 4f f3 | df 62 f7 a5

After MixColumns : 33 b0 58 fe | 57 a9 c5 bf | 05 ae 5e 87 | 51 bc bc be

After AddRoundKey : 45 66 ed d5 | ea b9 6f 11 | 49 90 14 24 | de 4c e1 90

e. Round 4

After SubBytes : 6e 33 55 03 | 87 56 a8 82 | 3b 60 fa 36 | 1d 29 f8 60

After ShiftRows : 6e 56 fa 60 | 87 60 f8 03 | 3b 29 55 82 | 1d 33 a8 36

After MixColumns : bc b7 77 de | 4e 57 09 0c | da 14 25 2e | f1 ae 3f d0

After AddRoundKey : 4e 2d d3 86 | 01 dd 27 fa | d9 a0 41 7b | 7d ea 06 ab

f. Round 5

After SubBytes : 2f d8 0d 44 | 7c c1 cc 2d | 35 e0 83 21 | ff 87 6f 62

After ShiftRows : 2f c1 83 62 | 7c e0 6f 44 | 35 87 0d 2d | ff d8 cc 21

After MixColumns : e7 e8 d8 7b | e8 52 8e 83 | d8 1a df 8f | 7b 3a c7 4c

After AddRoundKey : 1e c2 f0 cb | 5e 50 05 49 | 6d ac 30 10 | 42 c8 11 a8

g. Round 6

After SubBytes : 72 25 8c 1f | 58 53 6b 3b | 3c 91 04 ca | 2c e8 82 c2

After ShiftRows : 72 53 04 c2 | 58 91 82 1f | 3c e8 8c 3b | 2c 25 6b ca

After MixColumns : d7 1a 74 5e | 85 e3 f7 c5 | ec 43 9a 56 | 96 11 9a b5

After AddRoundKey : 87 64 b8 70 | 63 9f b0 21 | bf 89 32 2d | fc 29 e4 2a

h. Round 7

After SubBytes : 17 43 6c 51 | fb db f7 fd | 08 a7 23 d8 | b0 a5 69 e5

After ShiftRows : 17 db 23 e5 | fb a7 69 51 | 08 a5 6c fd | b0 43 e7 d8

After MixColumns : 9e 3a be 10 | 27 44 7d 7a | 75 10 69 30 | 81 dc 55 c4

After AddRoundKey : 89 b7 a9 3c | d6 b5 2d b2 | d7 2b 91 83 | 49 df d3 e8

i. Round 8

After SubBytes : a7 a9 d3 eb | f6 d5 d8 37 | 0e f1 81 ec | 3b 9e 66 9b

After ShiftRows : a7 d5 81 9b | f6 f1 66 eb | 0e 9e d3 37 | 3b a9 d8 ec

After MixColumns : 2b 15 dd 8b | 72 4e ed 5b | 41 70 74 31 | a2 ed 16 ff

After AddRoundKey : c7 dc bb 4f | 6f 76 db 57 | fe 73 ba 80 | d5 ed 5e 6c

k. Final Round

After SubBytes : df da a0 fe | 22 aa 64 a2 | 59 8a 19 4c | 38 ff bd d4

After ShiftRows : df aa 19 d4 | 22 8a bd fe | 59 ff a0 a2 | 38 da 64 4c

After AddRoundKey : 9d 56 21 66 | e9 d5 09 71 | a4 00 56 af | 84 85 98 50

Gambar III. 4. Gambar Asli Setelah di Enkripsi

Hasil CipherText adalah 9d 56 21 66 | e9 d5 09 71 | a4 00 56 af | 84 85 98 50

Analisis Dekripsi

CipherText

Block : 9d 56 21 66 | e9 d5 09 71 | a4 00 56 af | 84 85 98 50

Key : 2b 7e 15 16 | 28 ae d2 a6 | 08 08 08 08 | 08 08 08 08

Key Schedule

Round Key 1 : 7b e1 fd 3d | 1d c1 00 27 | 23 6f cc df | 37 dd c7 08

Round Key 2 : 2b 15 dd 8b | 72 4e ed 5b | 4170 74 31 | a2 ed 16 ff

Round Key 3 : 9e 3a be 10 27 44 7d 7a 75 10 69 30 81 dc 55 c4

Round Key 4 : d7 1a 74 5e 85 e3 f7 c5 ec 43 9a 56 96 11 9a b5

Round Key 5 : e7 4a 55 f7 e8 52 8e 83 d8 1a df 8f 7b 3a c7 4c

Round Key 6 : bcb7 77 de 4e 57 09 0c da 14 25 2e f1 ae 3f d0

Round Key 7 : 33 ba 58 fe 57 a9 c5 bf 05 ae 5e 87 51 bc bc be

Round Key 8 : 8a 8d 7a ab 6e 70 39 fb e6 ce 5e 24 2c1f fd d1

a. Initial Round

After AddRoundKey : df aa 19 d4 | 22 8a bd fe | 59 ff a0 a2 | 38 da 64 4c

b. Round 1

After InvShiftRow : df da a0 fe 22 aa 64 a2 59 8a 19 4c 38 ff bd d4

After InvSubByte : ef 7a 47 0c 94 62 8c 1a 15 cf 8e 5d 76 7d cd 19

After InvMixColumn : 94 9b ba 31 89 a3 8c 3d 36 a0 42 82 41 a0 0a 11

After AddRoundKey : c6 38 f4 50 a8 8f 58 84 bb 55 ea 5b 03 86 b9 19

c. Round 2

After InvShiftRow : c6 86 ea 84 a8 38 b9 5b bb 8f f4 19 03 55 58 50

After InvSubByte : c7 dc bb 4f 6f 76 db 57 fe 73 ba 8e d5 ed 5e 6c

After InvMixColumn : ec c9 66 c4 1d 38 36 0c bf 03 ce bf 77 00 48 93

After AddRoundKey : a7 d5 81 9b f6 f1 66 eb 0e 9e d3 37 3b a9 d8 ec

d. Round 3

After InvShiftRow : a7 a9 d3 eb f6 f5 d8 37 0e f1 81 ec 0e f1 81 ec 3b 9e 66 9b

After InvSubByte : c7 dc bb 4f 6f 76 db 57 fe 73 ba 8e d5 ed 5e 6c

After InvMixColumn : ec c9 66 c4 1d 38 36 0c bf 03 ce bf 77 00 48 93

After AddRoundKey : 17 db 23 e5 fb a7 69 51 08 a5 6c fd b0 43 e7 d8

e. Round 4

After InvShiftRow : 17 43 6c 51 fb db e7 fd 08 a7 23 d8 b0 a5 69 e5

After InvSubByte : 87 64 b8 70 63 9f b0 21 bf 89 32 2d fc 29 e4 2a

After InvMixColumn : 50 7e cc 2e e6 7c 47 e4 53 ca a8 7b 6a 38 7e 9f

After AddRoundKey : 72 53 04 c2 58 91 82 1f 3c e8 8c 3b 2c 25 6b ca

f. Round 5

After InvShiftRow : 72 25 8c 1f 58 53 6b 3b 3c 91 04 ca 2c e8 82 c2

After InvSubByte : 1e c2 f0 cb 5e 50 05 49 6d ac 30 10 42 c8 11 a8

After InvMixColumn : f9 88 a5 3c b6 02 8b ca b5 b6 ef 9f 39 f2 d6 e4

After AddRoundKey : 2f c1 83 62 7c e0 6f 44 35 87 0d 2d ff f8 cc 21

g. Round 6

After InvShiftRow : 2f d8 0d 44 7c c1 cc 2d 35 e0 8e 21 ff 87 6f 62

After InvSubByte : 4e 2d f3 86 01 dd 27 fa d9 a0 41 7b 7d ea 06 ab

After InvMixColumn : f2 9a 84 58 4f 8a 2e f6 03 b4 64 55 8c 44 39 7b

After AddRoundKey : 6e 56 fa 60 87 60 f8 03 3b 29 55 92 1d 33 a8 36

h. Round 7

After InvShiftRow : 6e 33 55 03 87 56 a8 82 3b 60 fa 36 1d 29 f8 60

After InvSubByte : 45 66 ed d5 ea b9 6f 11 49 90 14 24 de 4c e1 90

After InvMixColumn : 76 d6 b5 2b bd 10 aa ae 4c 3e 4a a3 8f f0 5d 2e

After AddRoundKey : 8f 4e ae 4a 06 e1 87 e4 f0 3e 4f f3 df 62 f7 a5

i. Round 8

After InvShiftRow : 8f 62 4f e4 06 4e ff7 f3 f0 e1 ae a5 df 3e 87 4a

After InvSubByte : 73 ab 92 ae a5 b6 26 7e 17 e0 be 29 ef d1 ea 5c

After InvMixColumn : f9 26 e8 05 cb c6 1f 85 f1 2e e0 0d c3 ce 17 8d

After AddRoundKey : 55 dc 12 4d 73 9e 87 b6 94 84 88 ca c4 32 2b c2

Berdasarkan perhitungan maka hasil dekripsi yang dihasilkan adalah :

32 43 f6 a8 | 88 5a 30 8d | 31 31 98 a2 | 04 04 04 04.

III.3. Desain Sistem

Desain sistem pada penelitian ini dibagi menjadi dua desain, yaitu desain sistem secara global untuk penggambaran model sistem secara garis besar dan desain sistem secara detail untuk membantu dalam pembuatan sistem. Pengembangan Model dengan Algoritma AES Dalam Enkripsi dan Dekripsi File Gambar dengan perancangan sebagai berikut :

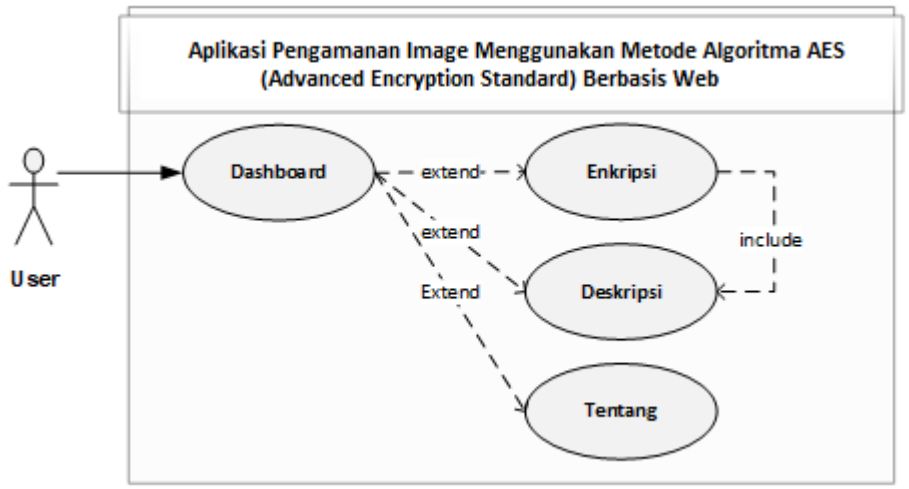
III.3.1. Desain Sistem Secara Global

Desain sistem secara global menggunakan bahasa pemodelan UML yang terdiri dari *Usecase Diagram*, *Acitvity Diagram* dan *Sequence Diagram*.

III.3.1.1. *Usecase* Diagram

Use case adalah rangkaian/uraian sekelompok yang saling terkait dan membentuk sistem secara teratur yang dilakukan atau diawasi oleh sebuah aktor. Use case digunakan untuk membentuk tingkah-laku benda/ things dalam sebuah model serta di Realisasikan oleh sebuah collaboration. Umumnya use case

digambarkan dengan sebuah elips dengan garis yang solid, biasanya mengandung nama. Use case menggambarkan proses system (kebutuhan system dari sudut pandang *user*) Maka digambarlah suatu bentuk diagram *Use Case* yang dapat dilihat pada gambar dibawah III.8. berikut :



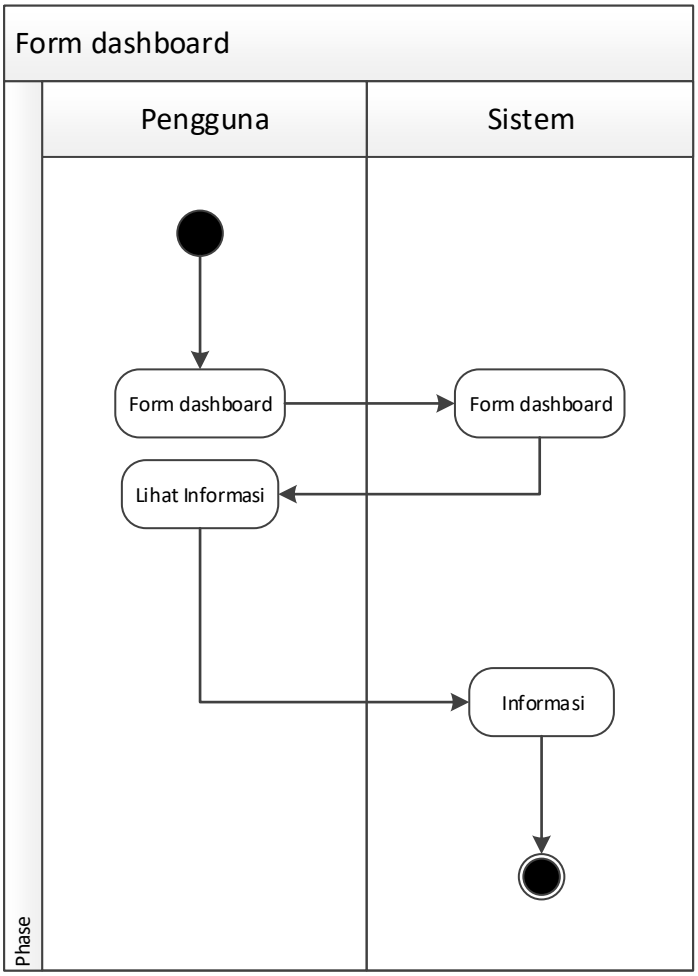
Gambar III. 5. *Use Case* Diagram Aplikasi Pengamanan *Image* Menggunakan Metode Algoritma *AES* (Advanced Encryption Standard) Berbasis Web

II*I.2.2.2. Acitvity Diagram*

Bisnis proses yang telah digambarkan pada *use case diagram* dijabarkan dengan *Activity diagram* :

1. *Activity Diagram Form Dashboard*

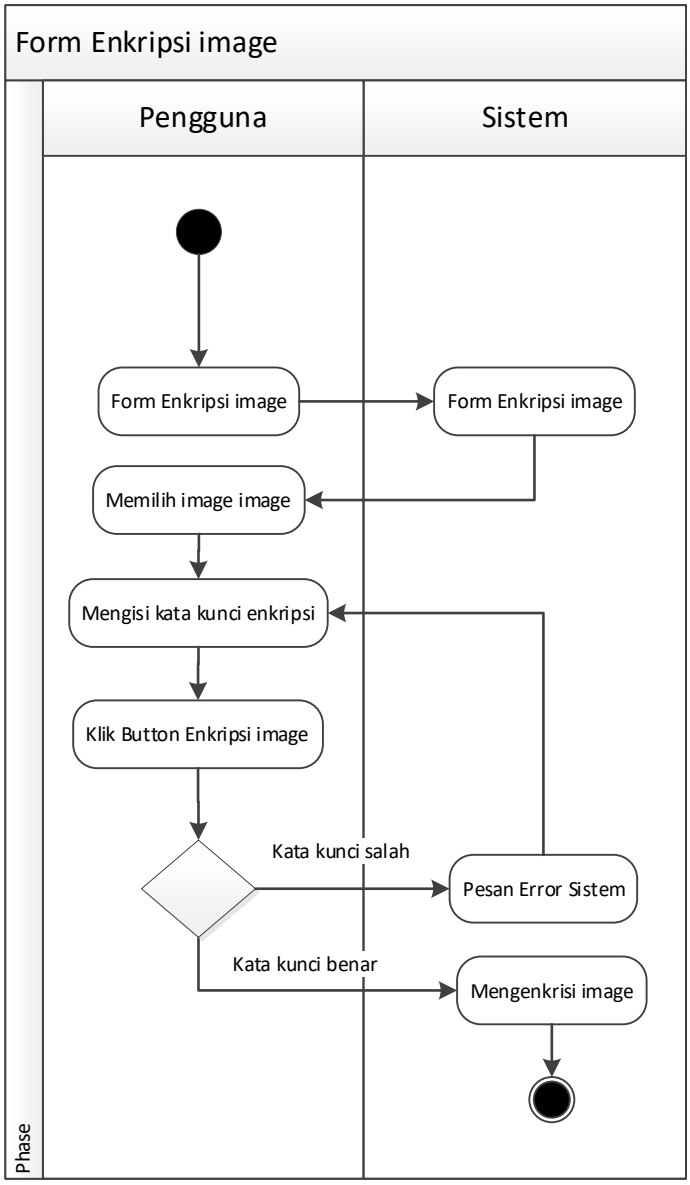
   Aktivitas yang dilakukan oleh *user* pada *form dashboard* dapat diterangkan dengan langkah-langkah *state* berikut, yang ditunjukkan pada gambar III.6 berikut :



Gambar III. 6. *Activity* Diagram *Form Dashboard*

2. *Activity Diagram Form Enkripsi*

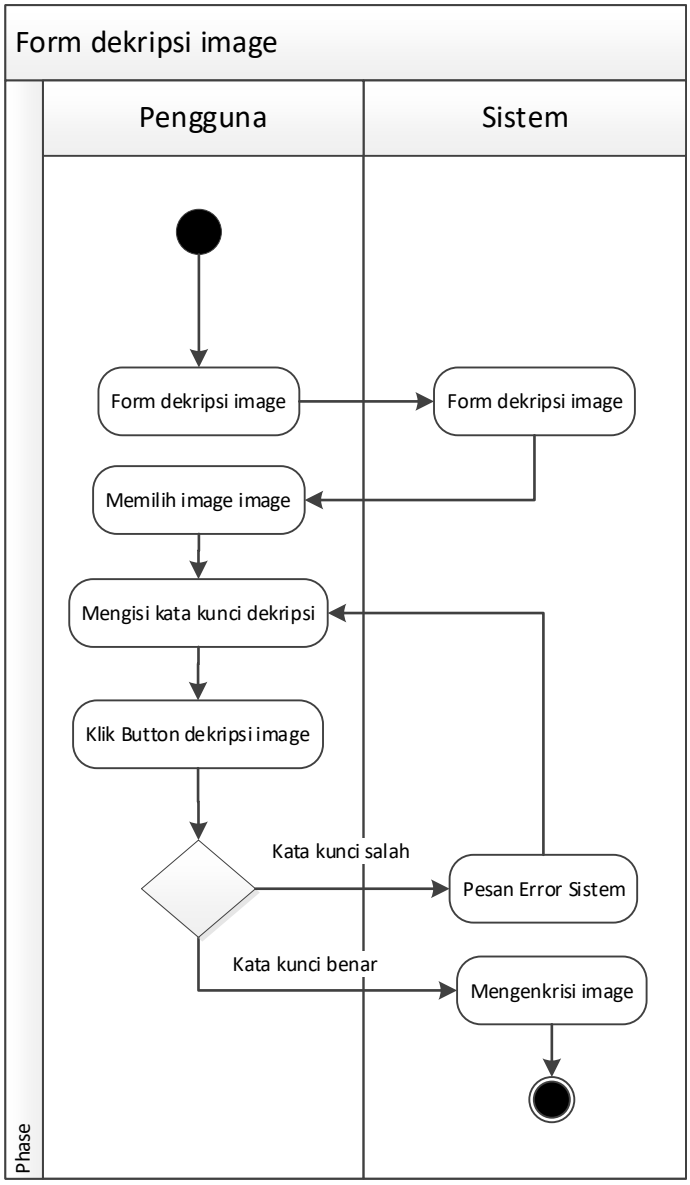
Aktivitas yang dilakukan oleh *user* pada *form Enkripsi* dapat diterangkan dengan langkah-langkah *state* berikut, yang ditunjukkan pada gambar III.7 berikut :



Gambar III. 7. *Activity* Diagram *Form* Enkripsi

3. *Activity Diagram Form Dekripsi*

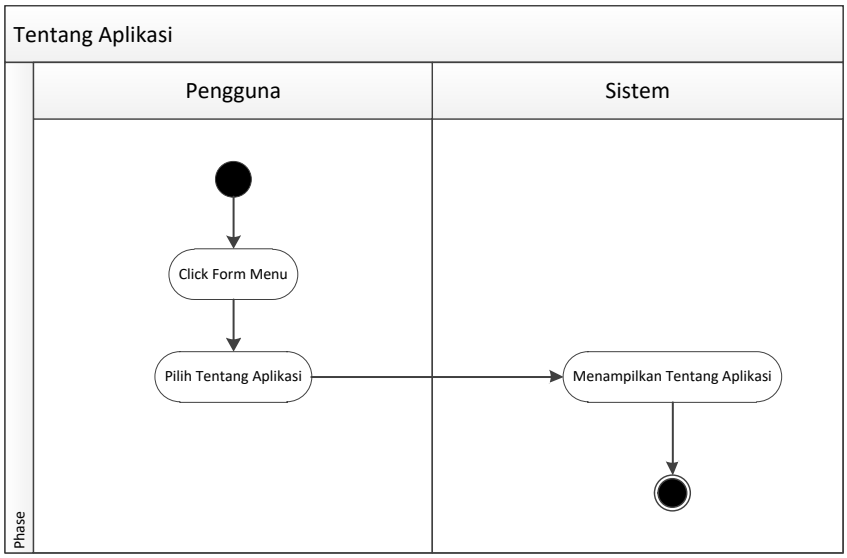
Aktivitas yang dilakukan oleh *user* pada *form Dekripsi* dapat diterangkan dengan langkah-langkah *state* berikut, yang ditunjukkan pada gambar III.8 berikut :



Gambar III. 8. *Activity* Diagram *Form* Dekripsi

4. *Activity Diagram* Tentang

Aktivitas yang dilakukan oleh *user* pada form tentang dapat diterangkan dengan langkah-langkah *state* berikut, yang ditunjukkan pada gambar III.9 berikut :



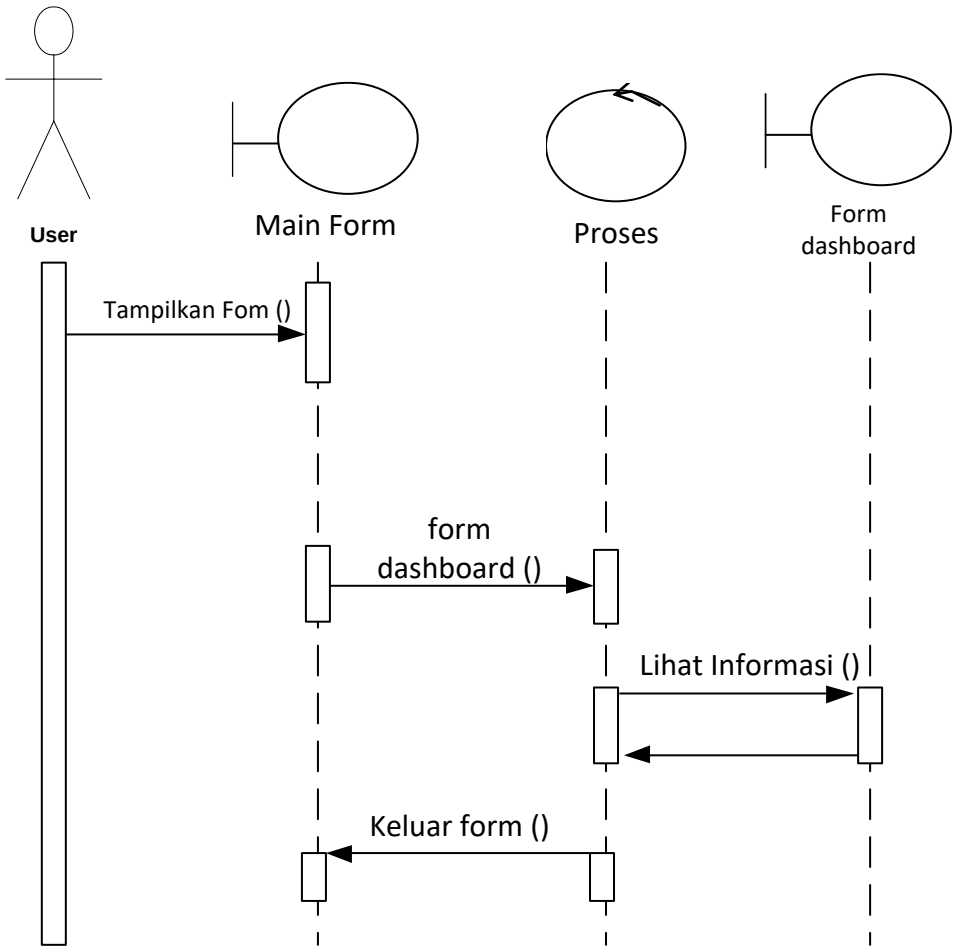
Gambar III. 9. *Activity* Diagram *Form* Tentang

III.2.2.3. *Sequence* Diagram

Sequence diagram adalah suatu diagram yang menggambarkan interaksi antar obyek dan mengindikasikan komunikasi diantara obyek-obyek tersebut. Diagram ini juga menunjukkan serangkaian pesan yang dipertukarkan oleh obyek – obyek yang melakukan suatu tugas atau aksi tertentu. Obyek – obyek tersebut kemudian diurutkan dari kiri ke kanan, aktor yang menginisiasi interaksi biasanya ditaruh di paling kiri dari diagram.Pada diagram ini, dimensi vertikal merepresentasikan waktu. Bagian paling atas dari diagram menjadi titik awal dan waktu berjalan ke bawah sampai dengan bagian dasar dari diagram. Garis Vertical, disebut lifeline, dilekatkan pada setiap obyek atau aktor. Kemudian, lifeline tersebut digambarkan menjadi kotak ketika obyek melakukan suatu operasi, kotak tersebut disebut activation box. Obyek dikatakan mempunyai *live activation* pada saat tersebut.Pesan yang dipertukarkan antar obyek digambarkan sebagai sebuah anak panah antara activation box pengirim dan penerima.

1. *Sequence Diagram Form Dashboard*

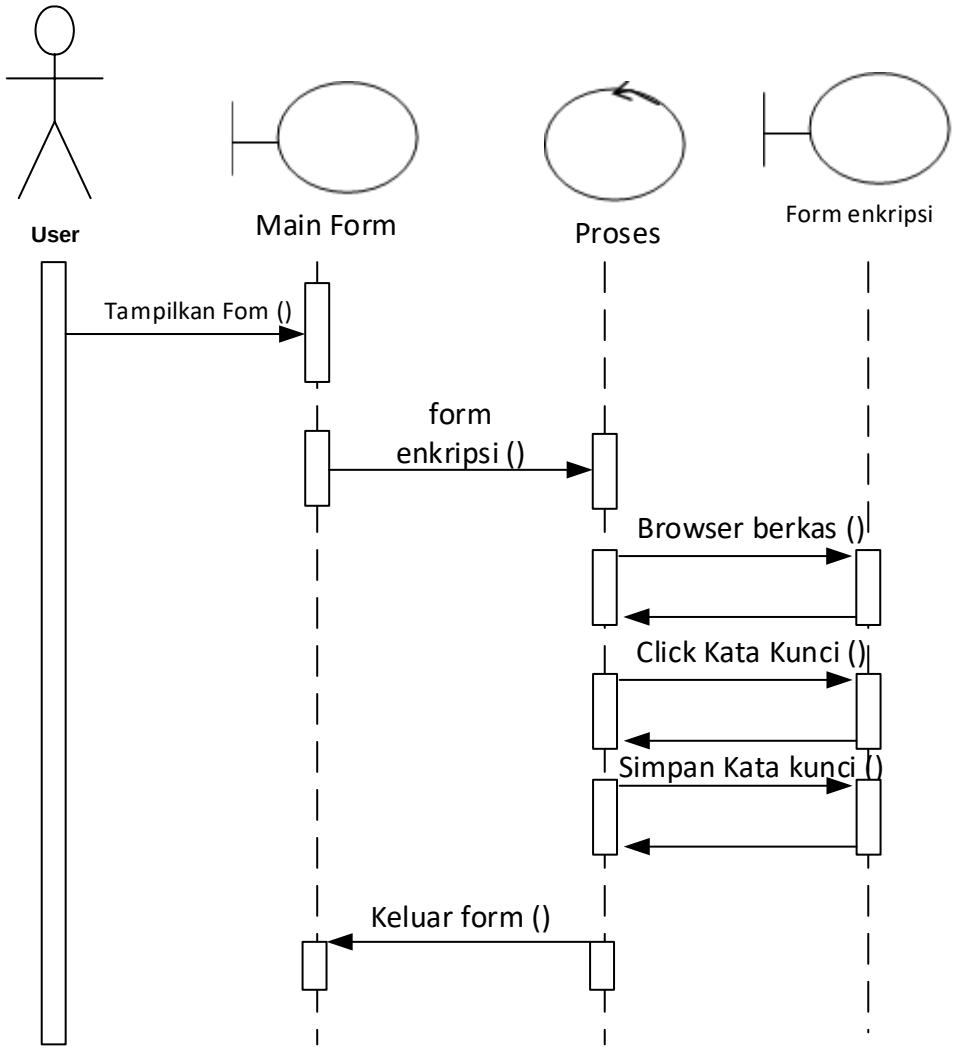
Aktivitas yang dilakukan oleh *user* pada *form Dashboard* dapat diterangkan dengan langkah-langkah *state* berikut, yang ditunjukkan pada gambar III.10 berikut :



Gambar III. 10. *Sequence Diagram Form Dashboard*

2. *Sequence Diagram Form Enkripsi*

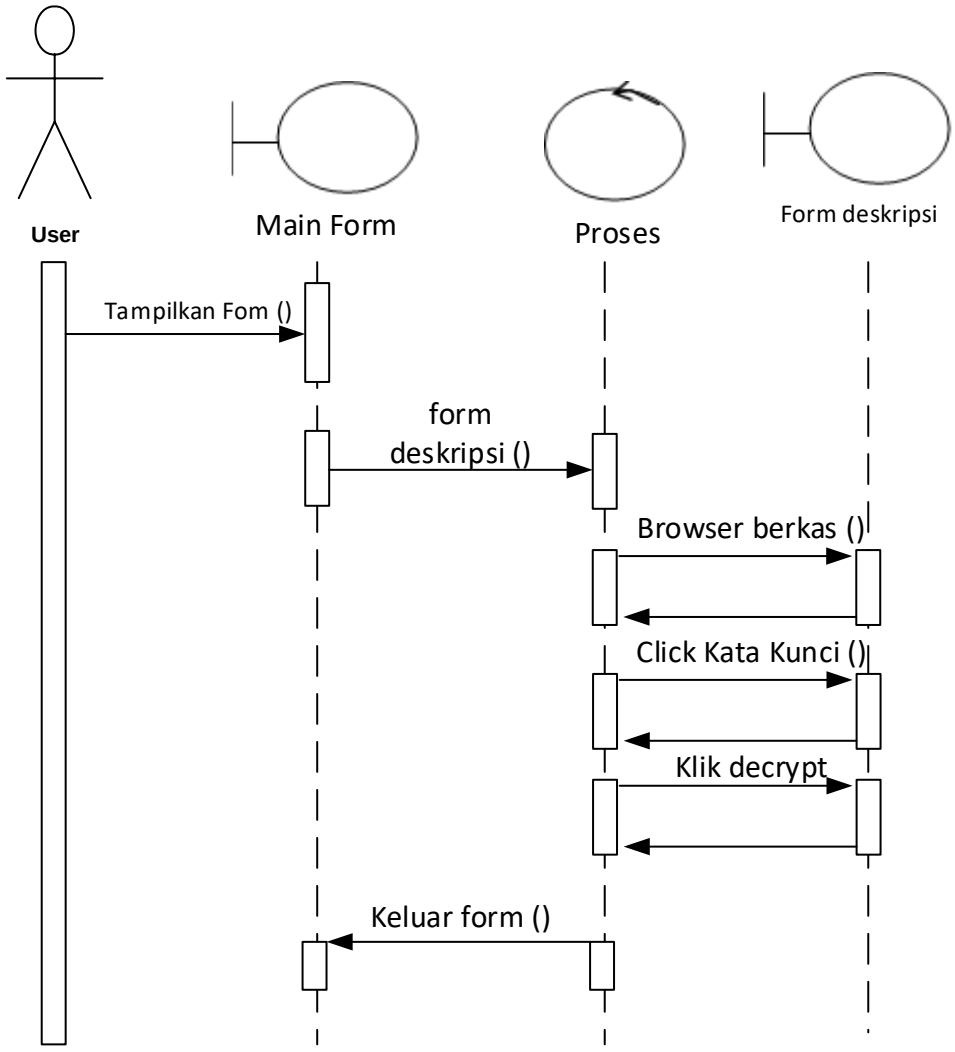
Aktivitas yang dilakukan oleh *user* pada *form Enkripsi* dapat diterangkan dengan langkah-langkah *state* berikut, yang ditunjukkan pada gambar III.11 berikut :



Gambar III. 11. *Sequence Diagram Form Enkripsi*

3. *Sequence Diagram Form Dekripsi*

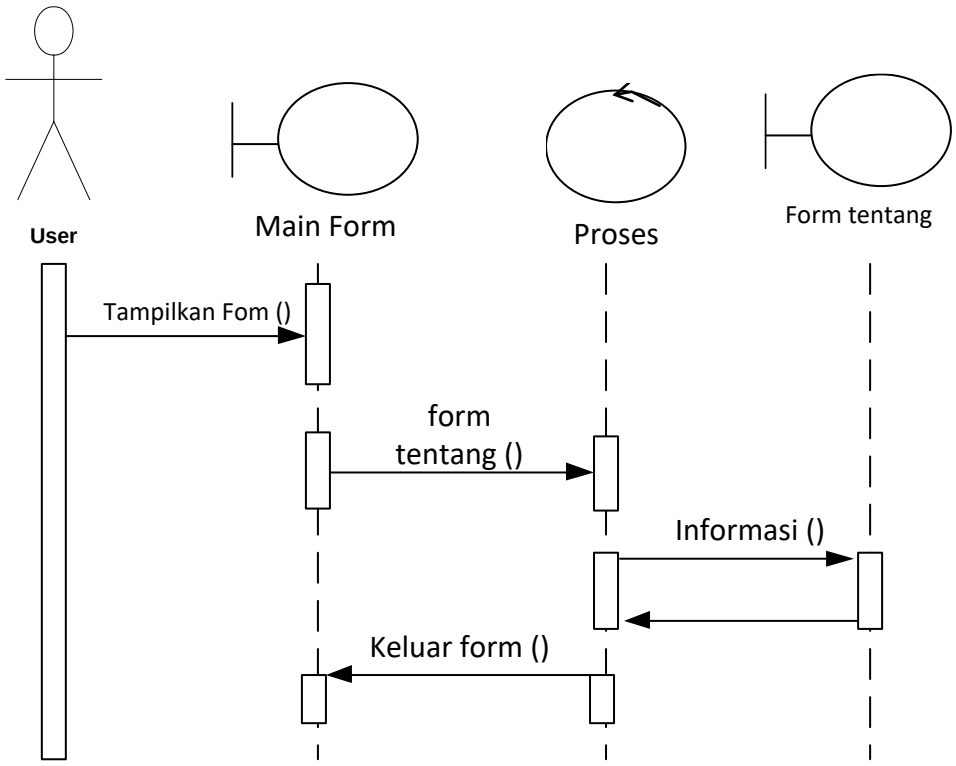
Aktivitas yang dilakukan oleh *user* pada *form Dekripsi* dapat diterangkan dengan langkah-langkah *state* berikut, yang ditunjukkan pada gambar III.12 berikut :



Gambar III. 12. ***Sequence Diagram Form Dekripsi***

4. *Sequence Diagram* Tentang

Aktivitas yang dilakukan oleh *user* pada form tentang dapat diterangkan dengan langkah-langkah *state* berikut, yang ditunjukkan pada gambar III.13 berikut :

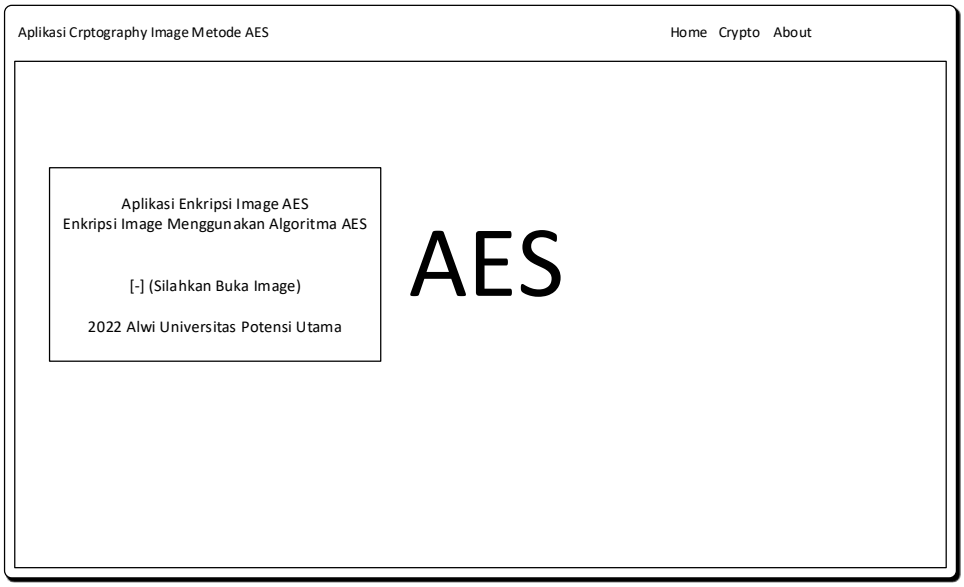


Gambar III. 13. ***Sequence Diagram Form Tentang***

III.3.3. Desain Sistem Secara Detail

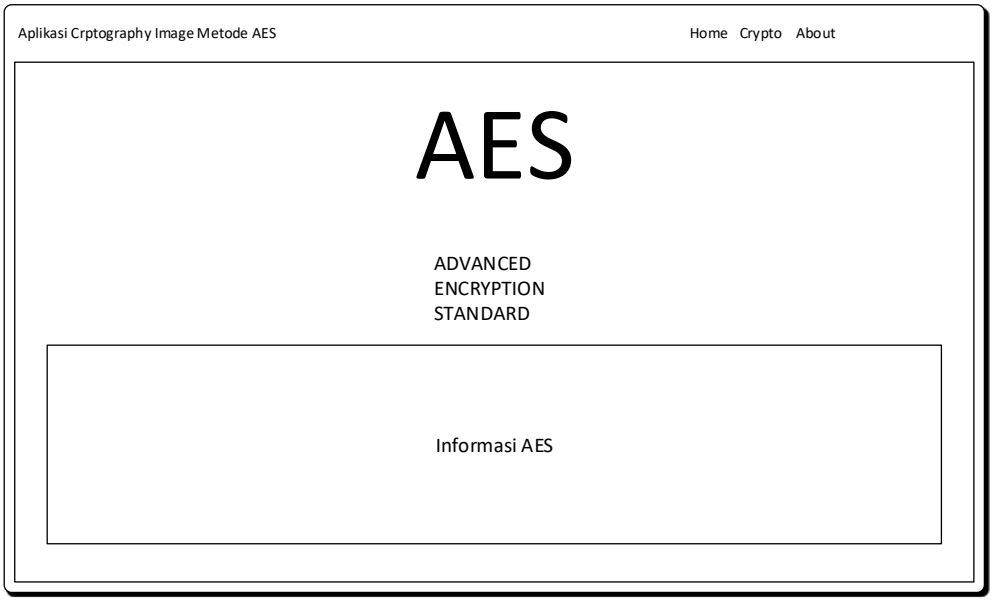
Tahap perancangan berikutnya yaitu desain sistem secara detail yang meliputi desain sistem.

1. *Form* Utama



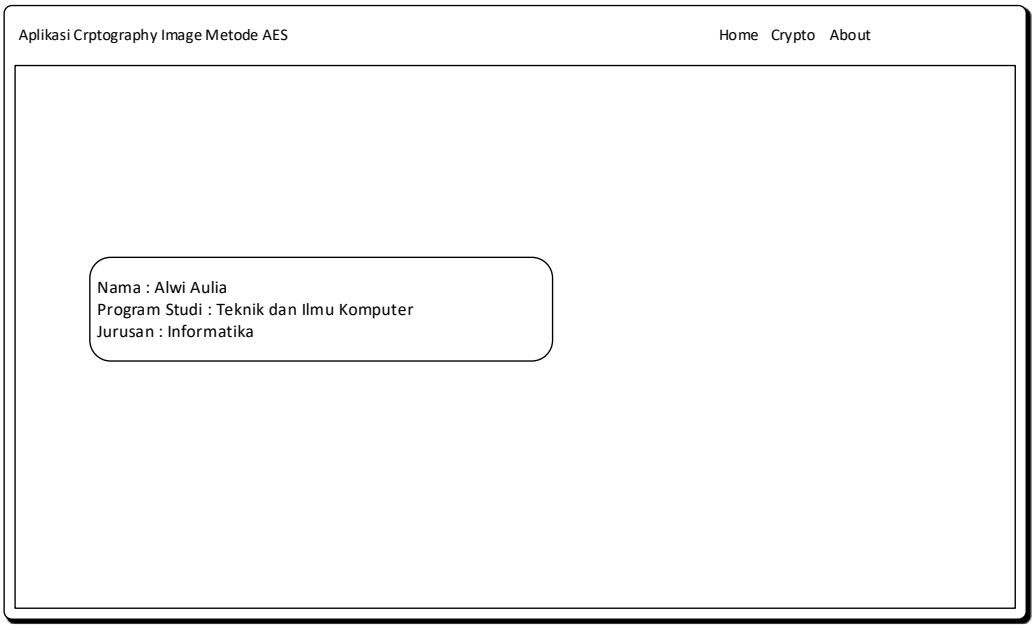
Gambar III. 14. *Form* Utama

2. *Desain Form* Tentang *AES*



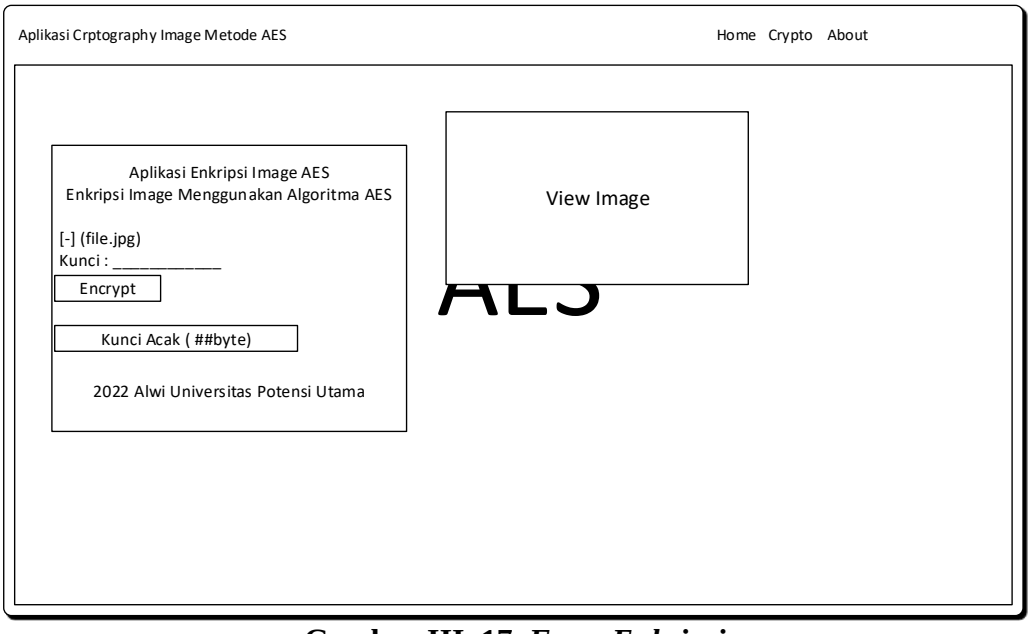
Gambar III. 15. Tampilan *menu* Tentang *AES*

3. *Desain Form About Me*

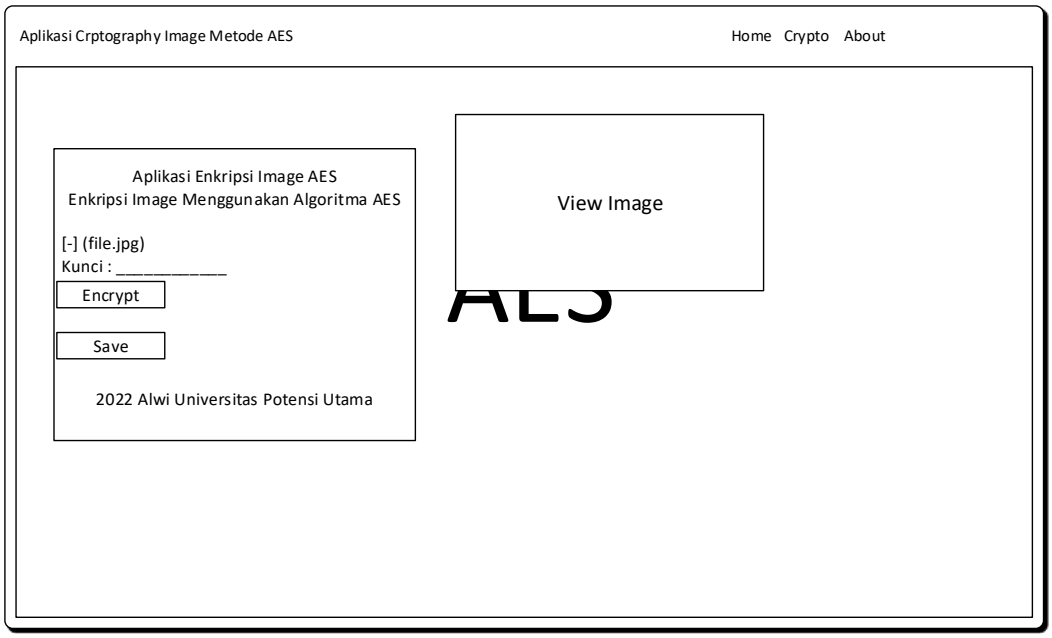


Gambar III. 16. Menu About.

4. *Desain* Form Enkripsi

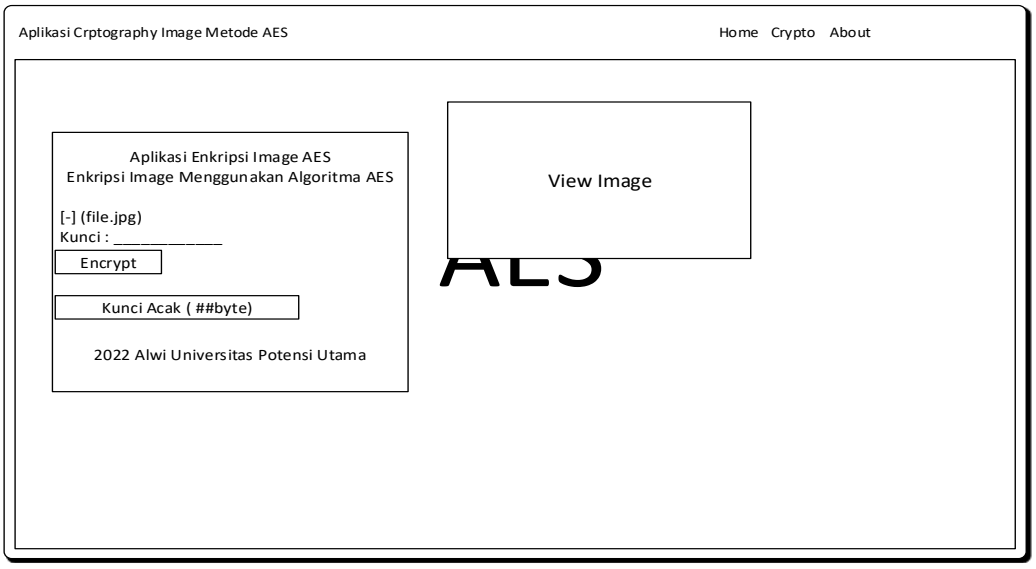


Gambar III. 17. *Form Enkripsi*



Gambar III. 18. *Form* Hasil Enkripsi

5. *Form* Dekripsi



Gambar III. 19. *Form* Dekripsi