BAB III

ANALISIS DAN DESAIN SISTEM

III.1. Analisis Masalah

Citra yang disimpan dalam komputer perlu dilindungi dari akses yang tidak diizinkan, kerusakan/perubahan yang merugikan. Bentuk-bentuk akses yang secara sengaja dapat merusak citra ataupun merugikan pemilik citra dapat berupa pengeditan citra (merubah bentuk asli), penghapusan dan penyebaran kepada publik sebagai bentuk kejahatan. Pengamanan citra merupakan aspek dalam jaringan yang mengacu upaya-upaya pengamanan dari akses yang merugikan tersebut.

Analisa sistem dilakukan untuk memahami informasi-informasi yang didapat dan dikeluarkan oleh sistem itu sendiri. Saat ini sistem komputer yang terpasang makin mudah diakses. Akses jarak jauh menyebabkan masalah keamanan menjadi salah satu kelemahan dalam komunikasi data. Karena hampir semua komunikasi data memungkinkan pengaksesan jarak jauh, maka perlu ada mekanisme pengamanan terhadap citra.

Untuk mengamankan citra, diperlukan kriptografi dengan metode enkripsi. Enkripsi merupakan salah satu cara yang dilakukan untuk mengamankan citra yang dikirimkan agar terjaga kerahasiaannya.

Salah satu teknologi metode enkripsi data yang digunakan adalah menggunakan algoritma *vigenere*, sehingga keamanan dan kerahasiaan data dapat terjaga saat melakukan pertukaran data.

III.1.1. Strategi Pemecahan Masalah

Strategi dalam melakukan pemecahan masalah yang dianalisa oleh penulis mengenai aplikasi keamanan data pada citra menggunakan algoritma *vigenere* dan adalah sebagai berikut:

1. Membuat aplikasi keamanan citra dengan menggunakan algoritma *vigenere*.
2. Meningkatkan keamanan dan privasi dalam citra saat pengguna melakukan komunikasi.
3. Mengetahui tingkat keamanan enkripsi data menggunakan algoritma *vigenere*.

III.1.2. Analisa Kebutuhan Fungsional

Kebutuhan fungsional adalah jenis kebutuhan yang berisi proses-proses apa saja yang nantinya dilakukan oleh sistem. Kebutuhan fungsional juga berisi informasi-informasi apa saja yang harus ada dan dihasilkan oleh sistem. Berikut merupakan kebutuhan fungsional yang terdapat pada aplikasi yang dibangun:

1. Mengimplementasikan penggunaan *VB .Net* dalam membuat aplikasi pengamanan citra menggunakan algoritma *vigenere*.
2. Aplikasi dapat melakukan enkripsi terhadap citra dengan berbagai jenis ekstensi gambar.
3. Aplikasi dapat melakukan dekripsi terhadap citra yang sudah dienkripsi.

III.1.3. Analisa Kebutuhan *Non*-Fungsional

Kebutuhan ini adalah tipe kebutuhan yang berisi properti perilaku yang dimiliki oleh sistem. Berikut adalah kebutuhan non-fungsional yang dimiliki sistem:

1. Operasional
    a. Dapat digunakan pada sistem operasi Windows XP/Vista/7 secara *stand alone*.
    b. Aplikasi yang dibangun menggunakan *Microsoft Visual Studio* 2010.
2. Kinerja

    Waktu yang diperlukan dalam mengeksekusi aplikasi sistem keamanan citra menggunakan algoritma *vigenere* yang dibangun cukup ringan sehingga eksekusi tampilannya cukup cepat.

III.2. Penerapan Algoritma

III.2.1 Algoritma *Vigenere Chiper*

Pada teknik substitusi *vigenere* setiap teks-kode bisa memiliki banyak kemungkinan teks-asli. Teknik dari substitusi *vigenere* bi sa dilakukan dengan dua cara, yaitu angka dan huruf. Teknik substitusi *vigenere* dengan menggunakan angka dilakukan dengan menukarkan huruf dengan angka, hampir sama dengan kode geser. Teknik substitusi *vigenere* dengan menggunakan huruf digunakan tabel *tabula recta* (disebut juga bujursangkar *vigenere*). *Tabula recta* digunakan untuk memperoleh teks-kode dengan menggunakan kunci yang sudah ditentukan.

Jika panjang kunci lebih pendek daripada pangjang teks-asli maka penggunaan kunci diulang.

Contoh:

A	B	C	D	E	F	G	H	I	J	K	L	M
0	1	2	3	4	5	6	7	8	9	10	11	12

N	O	P	Q	R	S	T	U	V	W	X	Y	Z
13	14	15	16	17	18	19	20	21	22	23	24	25

Gambar III.1 Contoh Tabel Subtitusi Algoritma Kriptografi Vigenere Cipher

Plaintext : PLAINTEXT

Kunci : CIPHER

Plain	15	11	0	8	13	19	4	23	19
Kunci	2	8	15	7	4	17	2	8	15
Hasil	17	19	15	15	17	10	6	5	8
Ciphertext	R	T	P	P	R	K	G	F	I

Gambar III.2 Contoh Tabel Kriptografi dengan Algoritma Vigenere Cipher

Dengan metode pertukaran angka dengan huruf di atas, diperoleh bahwa teks asli (PLAINTEXT) memiliki kode angka (15, 11, 0, 8, 13, 19, 4, 23, 19), sedangka kode angka untuk teks kunci (CIPHER) yaitu (2, 8, 15, 7, 4, 17, 2, 8, 15). Setelah dilakukan perhitungan, maka dihasilkan kode angka ciphertext (17, 19, 15, 15, 17, 10, 6, 5, 8). Jika diterjemahkan kembali menjadi huruf sesuai urutan awal , maka menjadi huruf RTPPRKGFI.

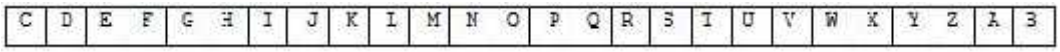
Sedangkan metode lain untuk melakukan proses enkripsi dengan metode vigenere cipher yaitu menggunakan *tabula recta* (disebut juga bujur sangkar *vigenere*).

	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

Gambar III.3 Contoh Tabula Recta Algoritma Kriptogrfi Vigenere Cipher

Kolom paling kiri dari bujursangkar menyatakan huruf-huruf kunci, sedangkan baris paling atas menyatakan huruf-huruf plaintext. Setiap baris didalam bujursangkar menyatakan huruf-huruf ciphertext yang diperoleh dengan caesar cipher, yang mana sejumlah pergeseran huruf plaintext ditentukan nilai

numerik huruf kunci tersebut (yaitu, a=0, b=1, c=2, ..., z=25). Sebagai contoh, huruf kunci c (=2) menyatakan huruf-huruf plaintext digeser sejauh 2 huru kekanan (dari susunan alfabetnya), sehingga huruf-huruf pada baris c adalah:



Gambar III.4 Potongan Tabula Recta baris ke - C

Bujursangkar vigenere digunakan untuk memperoleh ciphertext dengan menggunakan kunci yang sudah ditentukan. Jika panjang kunci lebih pendek daripada plaintext, maka kunci diulang penggunaanya (sistem periodik). Sebagai contoh, jika plaintext nya adalah THIS PLAINTEXT dan kunci adalah sony, maka penggunaan kunci periodik adalah sebagai berikut:

Plaintext : THIS PLAINTEXT

Kunci : sony sonysonys

Untuk mendapatkan ciphertext dari teks dan kunci diatas, untuk huruf plaintext pertama T, ditarik garis vertikal dari huruf T dan ditarik garis mendatar dari huruf s, perpotongannya adalah pada kotak yang berisi huruf L. Dengan cara yang sama, ditarik garis vertikal huruf H dan ditarik garis mendatar pada huruf o, perpotongannya adalah pada kotak yang juga berisi huruf V. Hasil enkrispsi seluruhnya adalah sebagai berikut:

Plaintext : THIS PLAINTEXT

Kunci : sony sonysonys

Ciphertext : LVVQ HZNGFHRVL

III.3. Perancangan

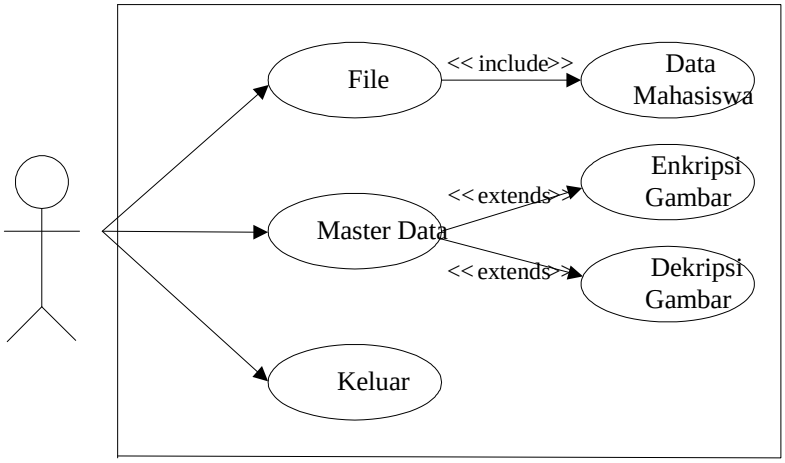
Desain sistem pada penelitian ini dibagi menjadi dua desain, yaitu desain sistem secara global untuk penggambaran model sistem secara garis besar dan desain sistem secara detail untuk membantu dalam pembuatan sistem.

III.3.1. Desain Sistem Secara Global

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

III.3.1.1. *Usecase* Diagram

Usecase diagram ini menggambarkan *user* (pengguna) yang menggunakan sistem dan perilaku *user* terhadap sistem yang terdapat pada Gambar III.5:



III.5 Diagram UseCase Aplikasi Keamanan Data Pada Citra

Pada Gambar III.5, merupakan *usecase* diagram dari aplikasi keamanan data pada *citra*. *User* yang didefenisikan pada aplikasi ini adalah orang yang menjalankan aplikasi. Ketika aplikasi dijalankan, aplikasi akan menampilkan

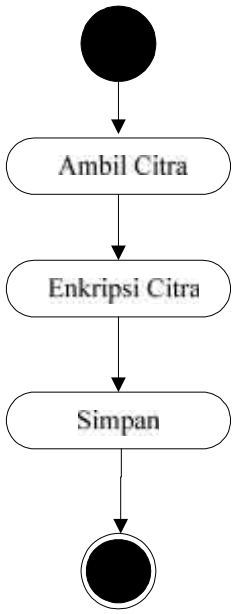
aplikasi enkripsi dan dekripsi citra. Setelah tampil, *user* dapat melakukan penginputan data pada kolom yang sudah tersedia dan mengeksekusi perintah.

III.3.2. *Activity Diagram*

Activity Diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis.

IIII.3.2.1. *Activity* Diagram Enkripsi

Berikut merupakan *activity* diagram enkripsi yang menjelaskan proses pengenkripsian pada citra:

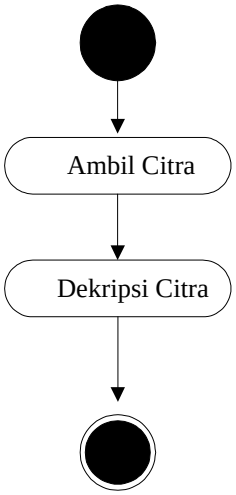


Gambar III.6 : *Activity* Diagram Enkripsi

Pada gambar *activity* diagram enkripsi, menunjukkan bahwa hal pertama yang harus dilakukan adalah pengambilan gambar. Setelah tampil, maka lakukan pengenkripsian terhadap dgambar yang diinput. Dalam hal ini pengenkripsian dilakukan dengan menggunakan algoritma *vigenere*. Kemudian data yang sudah dienkripsi dapat disimpan. Pada saat melakukan proses penyimpanan, data tersebut sudah dienkripsi dengan menggunakan algoritma *vigenere*.

III.3.2.2. *Activity Diagram* Dekripsi

Berikut merupakan *activity* diagram dekripsi yang menjelaskan proses pendekripsian pada database:

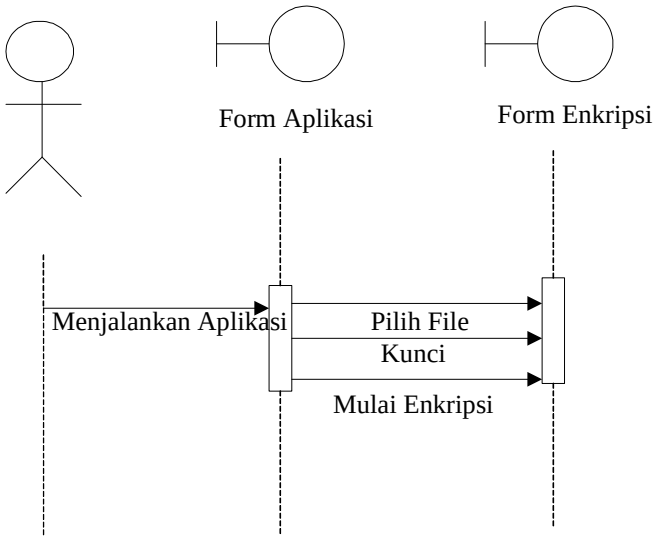


Gambar III.7 : *Activity* Diagram Dekripsi

Pada gambar *activity* diagram dekripsi, menunjukkan bahwa hal pertama yang harus dilakukan adalah pilih citra yang sudah dienkripsi. Kemudian dilakukan pendekripsian terhadap citra.

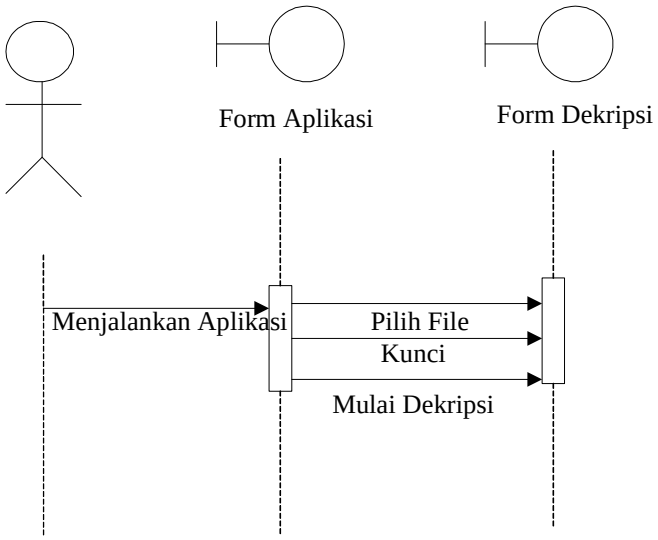
III.3.3. *Sequence Diagram*

Sequence Diagram menggambarkan kelakuan objek pada *usecase* dengan mendekripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek. Berikut merupakan *sequence* diagram yang menjelaskan sistem pengenkripsian citra.



Gambar III.8 : *Sequence* Diagram Proses Enkripsi

Pada gambar *sequence* diagram proses enkripsi menunjukkan, pertama-tama *user* menjalankan form aplikasi menu utam dan memilih form aplikasi Enkripsi, kemudian user memilih dan mengambil gambar yang akan dienkripsi. Kemudian user memasukkan kunci yang mana untuk merubah nilai keaslian (RGB) pada gambar. Kemudian user menjalankan proses enkripsi dean tombol enkripsi gambar.

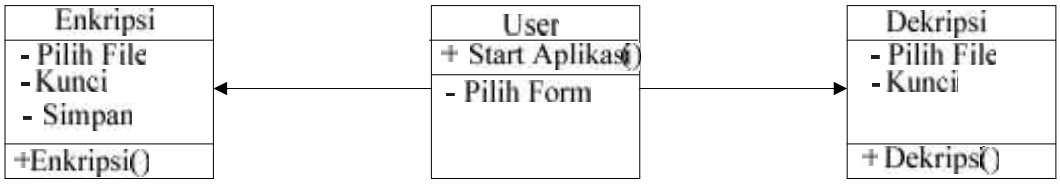


Gambar III.9 : *Sequence* Diagram Proses Dekripsi

Sequence diagram proses dekripsi menunjukkan bahwa seorang pengguna yang telah menjalankan program dapt langsung memilih *form* aplikasi dekripsi. Pengguna memilih *file* yang hendak di dekripsi. Pengguna memasukkan kunci untuk dapat melakukan dekripsi gambar.

III.4 Class Diagram

Class diagram menggambarkan keadaan suatu *system* (*attribbut*), dan memberikan pelayanan untuk menyelesaikan keadaan tersebut (metoda). Gambar 3.10 dibawah ini merupakan gambaran dari sebuah class diagram :



Gambar III.10 : *Class* Diagram Enkripsi dan Dekripsi Citra

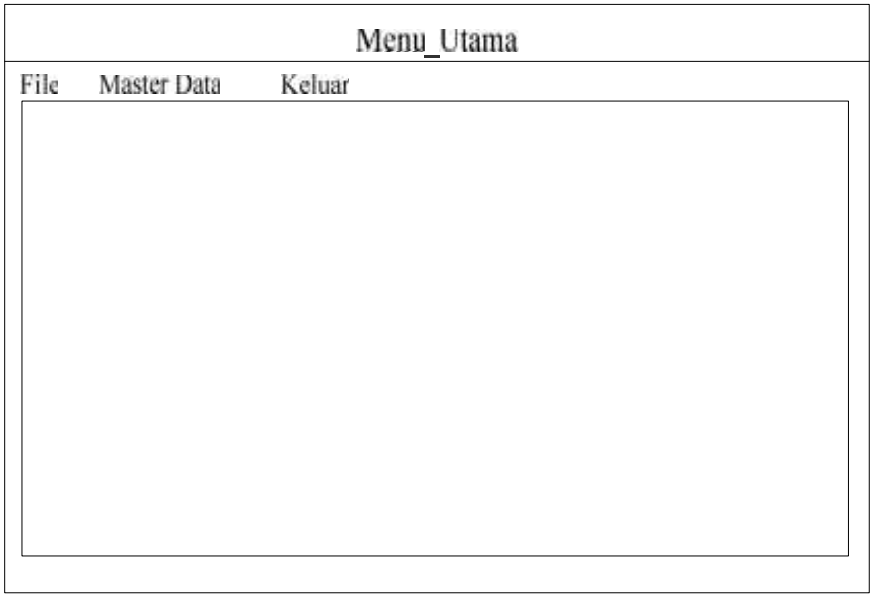
Pada gambar *class* diagram diatas, menunjukkan bahwa hal pertama yang harus dilakukan *user* adalah memulai menjalankan aplikasi, kemudian memilih form aplikasi yang tersedia. *User* dapat menjalankan form aplikasi enkripsi maupun dekripsi. Dalam *form* aplikasi enkripsi, user memilih file gambar yang hendak di enkripsi dan memasukkan kunci enkripsi. File yang selesai dieksekusi dapat disimpan di directory. Kemudian, user mampu melakukan dekripsi gambar yang sudah di enkripsi. Pilih *file* yang sudah di enkripsi dan memasukkan kode dekripsi, dimana kunci yang sama dengan kunci enkripsi. Kemudian jalankan program dekripsi.

III.5 Desain *User Interface*

Adapun *desain user interface* dalam perancangan aplikasi dibuat seperti pada gambar berikut:

III.5.1. Desain *User Interface* Menu Utama

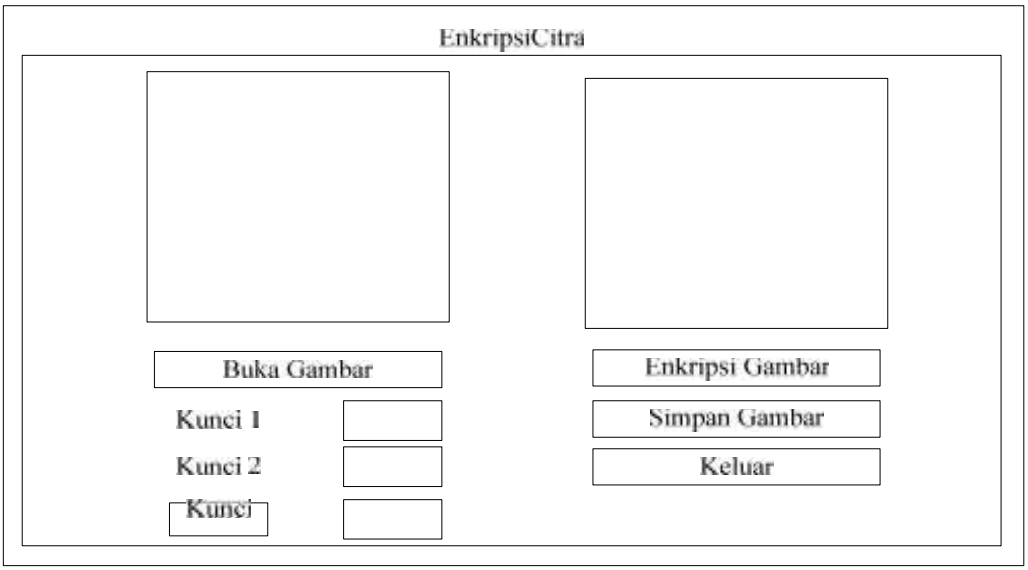
Perancangan *interface* untuk halaman utama perancangan aplikasi keamanan data pada database sebagai berikut:



Gambar III.11 : Tampilan Awal Aplikasi

III.5.2. Desain *User Interface Form* Enkripsi Citra

Form enkripsi citra merupakan *form* untuk melakukan proses enkripsi citra. Proses enkripsi menggunakan algoritma vigenere dan kemudian disimpan ke dalam *directori* kita pilih. Berikut tampilan *form* enkripsi citra:

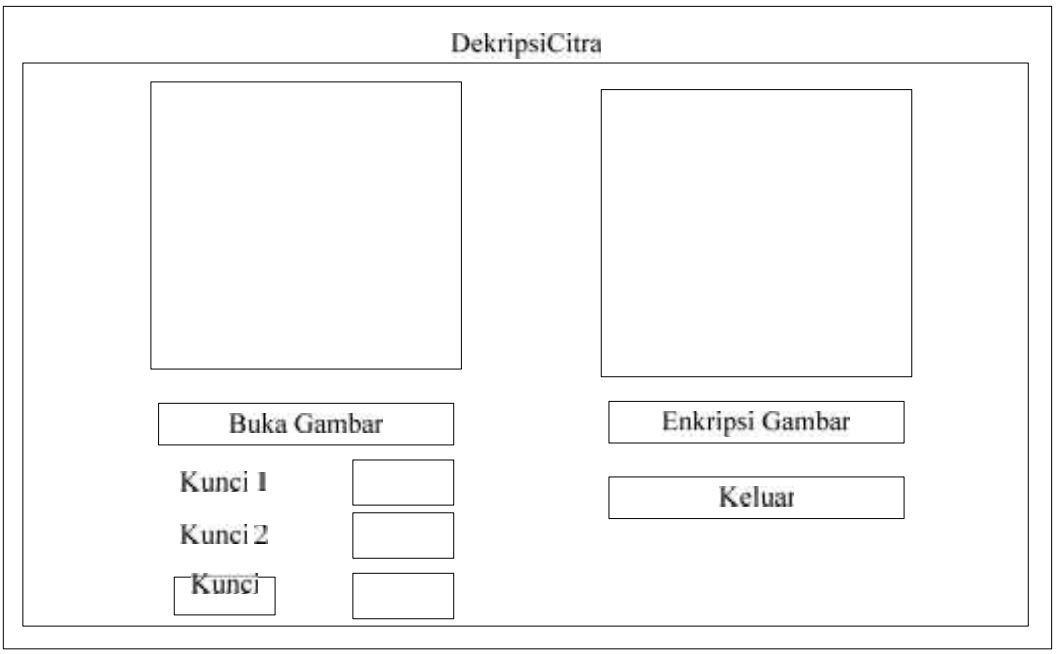


Gambar III.12 : Tampilan Enkripsi Citra

Pertama, lakukan penginputan gambar dengan tekan tombol buka gambar, kemudian masukkan kunci 1 dan kunci 2, kemudian tekan tombol kunci proses enkripsi menggunakan algoritma *vigenere* dengan cara klik tombol enkripsi gambar . Data yang sudah di enkripsi dapat disimpan dan dilihat di directori yang dipilih.

III.5.3. Desain *User Interface Form* Dekripsi Citra

Form dekripsi citra merupakan *form* untuk melakukan proses dekripsi citra. Proses dekripsi menggunakan algoritma *vigenere*. Berikut tampilan *form* dekripsi *citra*:



Gambar III. 13: Tampilan Dekripsi Citra

Lakukan hal yang sama dengan enkripsi yaitu masukkan gambar pada *picture box* yang sudah disediakan, kemudian lakukan proses dekripsi menggunakan algoritma *vigenere* dengan cara klik tombol dekripsi gambar. Klik tombol keluar untuk keluar dari aplikasi.