BAB III

ANALISIS DAN DESAIN SISTEM

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 adalah dengan menggunakan metode steganografi. Hal ini dikarenakan metode steganografi sangat mudah diimplementasikan. Meskipun steganografi adalah salah satu cara untuk mengamankan data, namun masih ada kekurangannya yaitu metode steganografi telah diketahui oleh banyak orang. Karena itu dibutuhkan suatu gabungan metode, untuk mendapatkan keamanan lebih dalam informasi atau data, khususnya yang bersifat rahasia.

III.1.2 Analisis Spesifikasi

Dalam pembuatan aplikasi ini dibutuhkan spesifikasi dasar, antara lain:

1. Menggunakan *intel processor dualcore*.
2. Menggunakan *memory* 1 GB.
3. Menggunakan *windows xp sp3*.
4. Program aplikasi dibuat menggunakan *Visual Basic 2010*.
5. Visual Studio 2010 membutuhkan *.NET framework versi 4.*
6. Metode yang digunakan pada proses kriptografi adalah *Hill Cipher*
7. Data yang dienkripsi adalah teks.
8. Proses steganografi menggunakan metode LSB .

9. Media file yang disisipkan berekstensi *. bmp.

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

1. Menggunakan *intel processor pentium 4* atau yang lebih tinggi
2. Sistem membutuhkan Sistem Operasi *Windows XP* atau *Windows 7* atau Windows 8 yang compatible dengan berbagai aplikasi yang ada.
3. Memori minimal 256 MB untuk *Windows XP*, 1 GB untuk *Windows 7*, dan 1GB untuk *Windows 8*.

III.2. Penerapan Metode

Metode yang digunakan di dalam penelitian ini adalah metode penggabuangan antara *algoritma hill cipher* dan steganografi LSB *(Least significant bit).*

1. Algoritma *hill cipher*

Hil Cipher termasuk dalam salah satu kripto sistem polialfabetik, artinya setiap karakter alfabet bisa dipetakan ke lebih dari satu macam karakter alfabet. Cipher tersebut ditemukan pada tahun 1929 oleh Lester S. Hill. Ide dari Hill Cipher adalah misalkan m adalah bilangan bulat positif, Dengan cara mengambil m kombinasi linier dari m karakter alfabet dalam satu elemen plaintext. Misalkan m=2, maka kita dapat menuliskan suatu elemen plaintext sebagai x = (x1 , x2) dan suatu elemen ciphertext sebagai y = (y1 , y2). Disini (y1 , y2) adalah kombinasi linier dari x1 dan x2 (Dony Ariyus:34,2006).

Kita misalkan:

$$Y_1 = 1x_1 + 5x_2$$
$$Y_2 = 9x_1 + 8x_2$$

..... **(1)**

Sehingga dapat dituliskan kedalam bentuk matrik sebagai berikut:

$$Y_{1,}\, Y_2 = X_{1,}\, X_2 = \begin{pmatrix} 1 & 5 \\ 9 & 8 \end{pmatrix}$$

..... **(2)**

Secara umum, algoritma Hill Cipher akan menggunakan matrik K m x m sebagai kunci untuk mengacak pesannya . Jika elemen pada baris i dan kolom j dari matriks Kij, maka dapat dituliskan sebagai berikut:

Dikatakan bahwa ciphertext diperoleh dari plaintext dengan cara transformasi linier. Untuk melakukan dekripsi, akan digunakan matrik invers K-1. Jadi, dekripsi dilakukan jika matrik tersebut memiliki nilai invers dengan rumus x = yK-1.

a. Perkalian matriks memiliki sifat asosiatif, yaitu (AB)C = A(BC).
b. Matriks invers dari A adalah A-1 dimana AA-1 = A-1A = Im.
c. Matriks identitas m x m yang ditulis dengan Im , adalah matrik yang berisi 1 pada diagonal utama dan berisi 0 pada elemen lainnya. Contoh matrik identitas 2 x 2:

$$I_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad \text{.....} \quad \textbf{(3)}$$

Im disebut dengan matrik identitas karena AIm = A Untuk sembarang matrik I x m dan ImB = B untuk sembarang matrik m x n.

Dengan menggunakan sifat-sifat matrik diatas, maka dapat ditentukan matriks kunci dan matrik kunci invers sebagai berikut :

$$K = \begin{pmatrix} 1 & 5 \\ 9 & 8 \end{pmatrix} \quad \text{.....} \quad \textbf{(4)}$$

$$K^{-1} = \begin{pmatrix} 4 & 17 \\ 15 & 7 \end{pmatrix} \quad \text{.....} \quad \textbf{(5)}$$

Setelah ditetapkan kunci *enkripsi* dan pengembalian pesan maka terdapat lagi rumus yang digunakan untuk mengubah huruf asli kebentuk kode dapat dilihat pada persamaan (6) dan rumus pengembalian pesan dapat dilihat pada persamaan (7) sebagai berikut:

$$C=(P*K) \bmod 26 \quad \text{.....} \quad \textbf{(6)}$$

$$P = K^{-1} * C \quad \text{.....} \quad \textbf{(7)}$$

2. Steganografi LSB (Least significant bit)

Steganografi digunakan untuk menyisipkan sebuah teks kedalam media digital yang bisa berupa gambar, audio, atapun video. Sedangkan perbedaan dengan watermarking adalah terletak pada data yang akan disisipkan berupa gambar. Steganografi berasal dari bahasa Yunani yaitu Steganos yang berarti menyembunyikan dan Graptos yang artinya tulisan sehingga secara keseluruhan artinya adalah tulisan yang disembunyikan. Secara umum steganografi merupakan seni atau ilmu yang digunakan untuk menyembunyikan pesan rahasia dengan segala cara sehingga selain orang yang dituju, orang lain tidak akan menyadari keberadaan dari pesan rahasia tersebut. Steganografi sudah digunakan sejak dahulu kala sekitar 2500 tahun yang lalu untuk kepentingan politik, militer, diplomatik, serta untuk kepentingan pribadi sebagai alat (Stefanus:8,2004). Steganografi mempunyai 2 metode, yaitu MSB *(Most Significant Bit)* dan LSB *(Least Significant Bit)*. Untuk MSB adalah sebuah metode penyisipan dengan cara mengganti nilai bit-bit atas dari data media penampung dengan nilai bit-bit dari data yang akan disembunyikan. Sedangkan LSB adalah kebalikan dari MSB, yaitu sebuah metode menyembunyikan data pada bit bawah (LSB) pada data *pixel* yang menyusun *file* tersebut. (William Stalling:56,2003). Seperti untuk *file* bitmap 24 bit maka setiap *pixel* (titik) pada gambar tersebut terdiri dari susunan tiga warna merah,

hijau dan biru (RGB) yang masing-masing disusun oleh bilangan 8 bit (byte) dari 0 sampai 255 atau dengan format biner 00000000 sampai 11111111. Dengan demikian pada setiap *pixel file* bitmap 24 bit kita dapat menyisipkan 3 bit data. Contohnya huruf A dapat disisipkan dalam 3 *pixel*,

01100111	11010011	11001000
R	G	B

..... **(8)**

Sedangkan representasi biner huruf A adalah 01100001. Dengan menyisipkan 4 bit kedalam nilai R dan 4 bit kedalam nilai B pada *pixel* diatas dengan menggunakan metode LSB maka akan dihasilkan:

Rawal = 00100111, setelah disisipkan 4 bit dari kiri 0110.

Maka Rbaru = 00100110.

Bawal = 11001000, setelah disisipkan 4 bit dari kiri 0001.

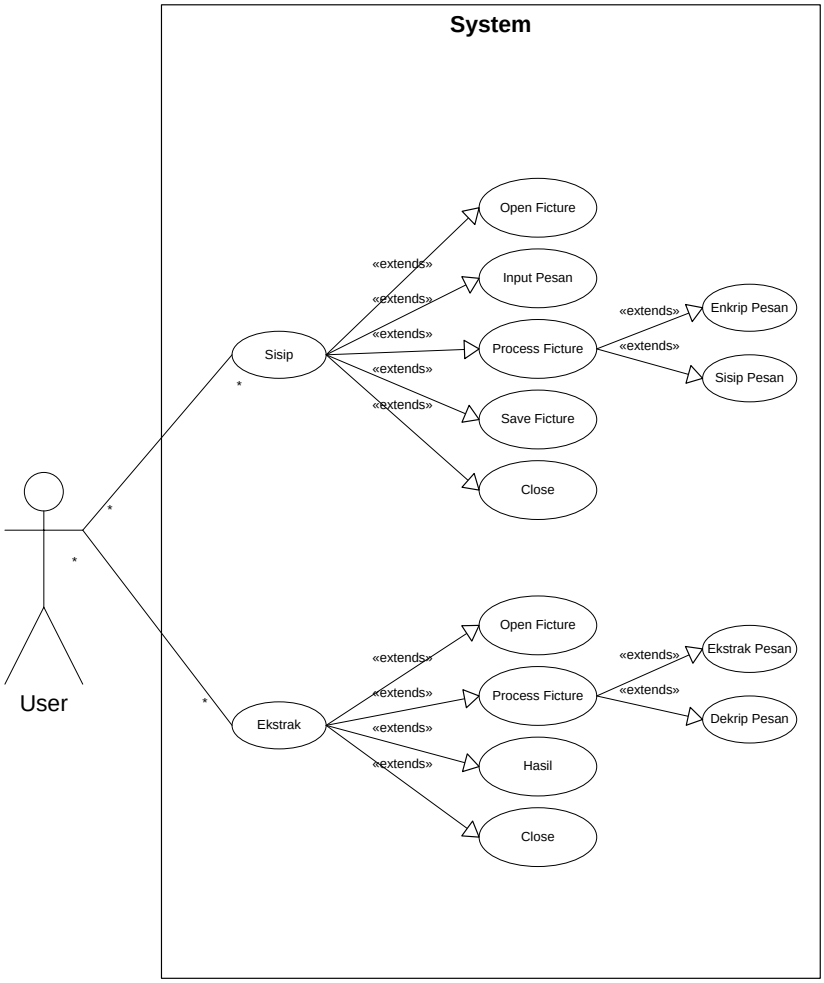
Maka Bbaru = 11000001.

Terlihat hanya empat bit rendah yang berubah, sehingga untuk mata manusia tidak akan tampak perubahannya.

III.3. Desain Sistem

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*. *User* memiliki dua *use case*, yaitu *form* sisip dan *form* ekstrak dan yang kemudian keduanya memilki beberapa *use case* lagi. *Use case diagram* perangkat lunak yang dibangun terlihat pada gambar III.1 berikut:



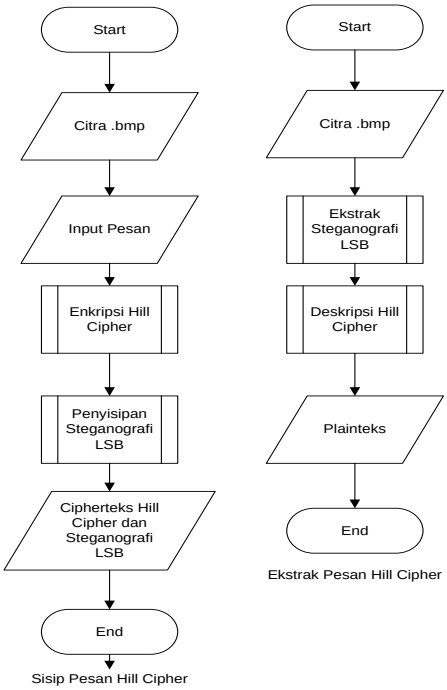
Gambar III.1 *Use Case* Algoritma *Hill cipher* Dan LSB (*Least Significant Bit*)

III.3.2 *FlowChart* Program

Pada gambar sebelumnya, terdapat proses yaitu sisip, dan ekstrak. Jika pengguna ingin memakai aplikasi penyisipkan sebuah teks ke stegano, maka dilanjutkan dengan proses enkripsi yang menghasilkan cipherteks kemudian dilanjutkan proses penyisipan yang menghasilkan stegano image. Jika pengguna ingin memakai aplikasi mengekstrak pesan dari citra. proses ekstraksi dengan memasukkan citra hasil steganografi sehingga menghasilkan cipherteks, dapat dilihat pada gambar III.2 berikut.

1. ***Flowchart* Program**

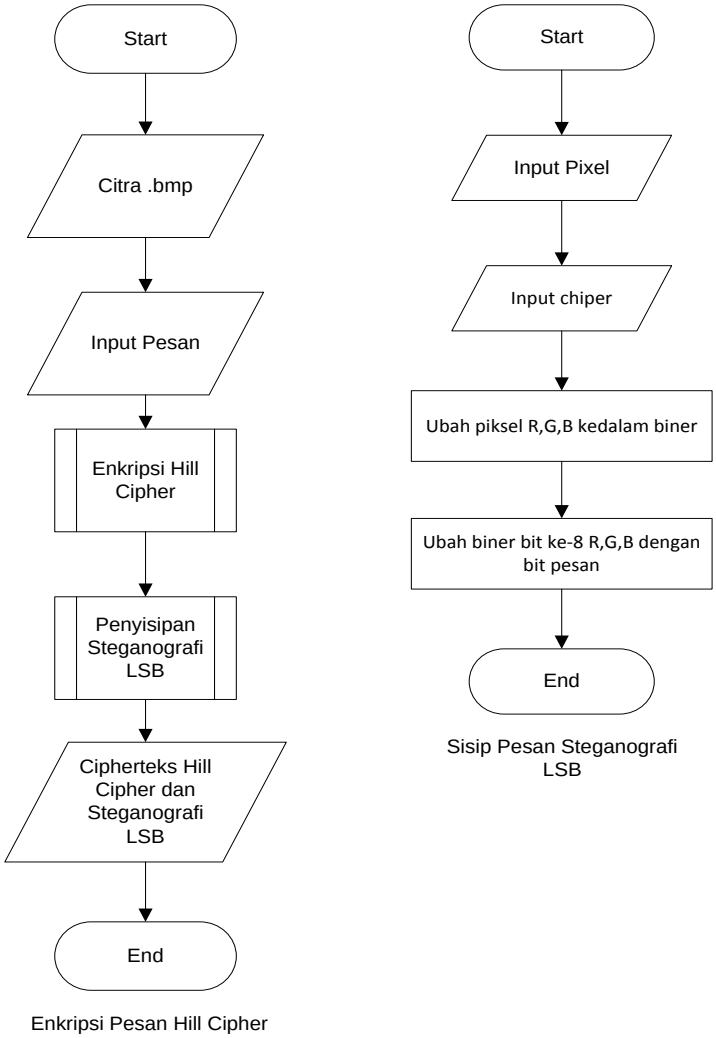
Flowchart program merupakan suatu gambaran atau proses jalannya suatu program secara *global* yang dijelaskan melalui gambar III.2 dibawah ini:



Gambar III.2 *Flowchart Penyisipan Pesan Dan Ekstrak Pesan*

2. ***Flowchart* Sisip Pesan**

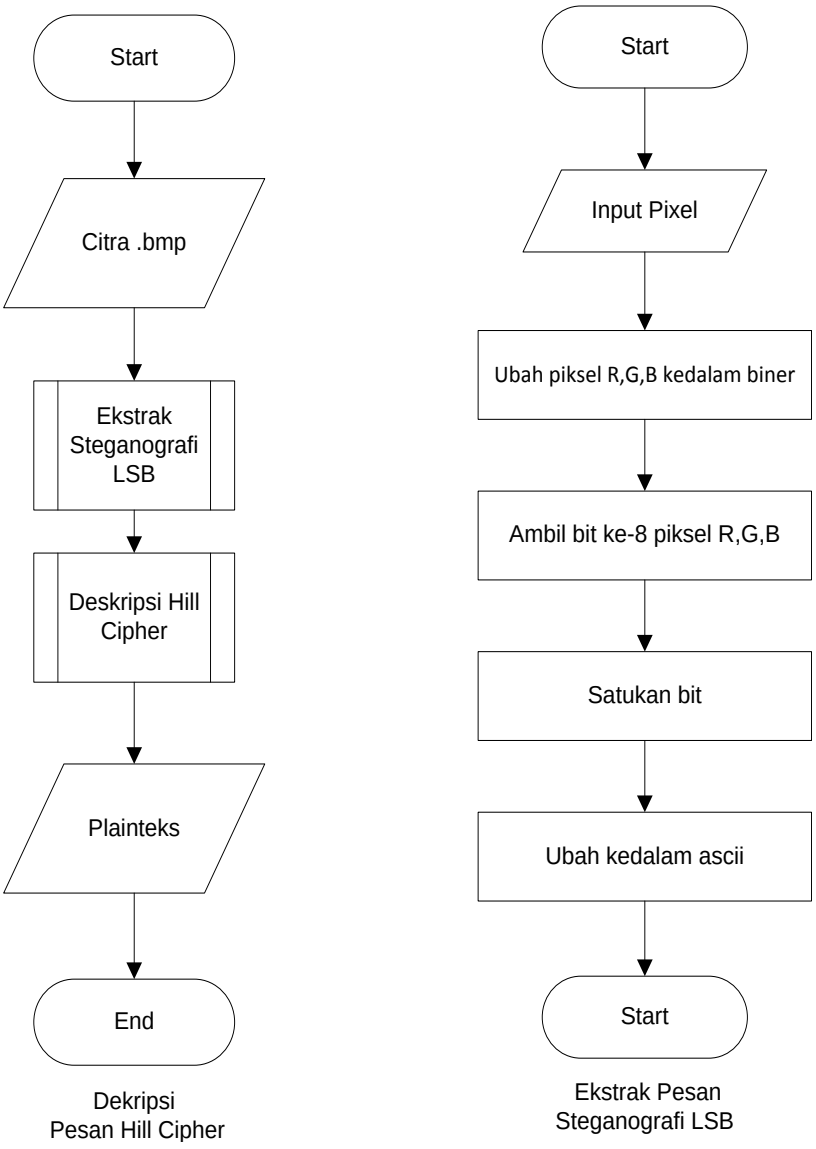
Flowchart sisip pesan merupakan suatu proses mengubah data teks *(enkripsi)*menggunakan algoritma *Hill Cipher* dan menyisipkan pesan mengggunakan steganografi LSB. Proses tersebut dapat dilihat pada gambar III.3 dibawahini :



Gambar III.3 *Flowchart* Sisip Pesan

3. Flowchart Ekstrak Pesan

Flowchart ekstrak pesan merupakan suatu proses pengembalian data teks asli (*plainteks)* dan citra kebentuk semula. Proses tersebut dapat dilihat pada gambar III.4 dibawahini :



Gambar III.4 *Flowchart* Ekstrak Pesan

III.4. Desain User Interface

1. Antarmuka *Form* Utama

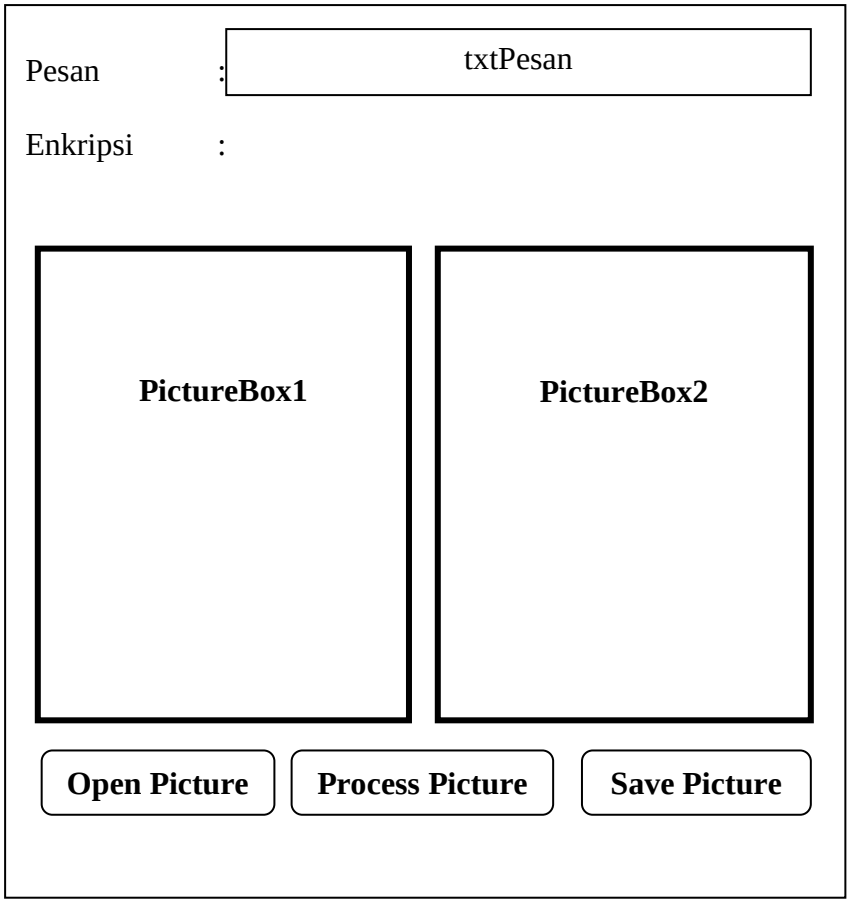
Rancangan *form* ini dibuat sebagai *form* utama dimana di *form* ini ada terdapat dua *button* yang akan membuka *form* lain seperti *form* penyisipan pesan, ekstrak pesan dan bantuan, dapat dilihat pada gambar III.5 dibawah ini :

Gambar III.5 Tampilan *Form* Utama

2. Antarmuka *Form* Sisip Pesan

Rancangan *form* ini dibuat untuk melakukan proses penyembunyian pesan dimana akan dijelaskan satu persatu tentang *form* tersebut. Berikut penjelasan *form* penyembunyian pesan atau dapat dilihat pada gambar III.6 dibawah ini:

a. Txtpesan merupakan tempat dimana menulis kalimat/pesan yang ingin disembunyikan.
b. Enkripsi merupakan tempat untuk menampilkan hasil enkripsi
c. *Picturebox*1 (sebelah kiri) untuk memunculkan gambar yang ingin disisipkan pesan.
d. *Picturebox2* (sebelah kanan) untuk memunculkan gambar hasil proses enkripsi berganda dan LSB.
e. Tombol/*button process picture* merupakan tombol untuk mengeksekusi pesan teks yang ingin disisipkan.
f. Tombol/*button save picture* merupakan tombol untuk menyimpan gambar yang telah disisipkan teks.



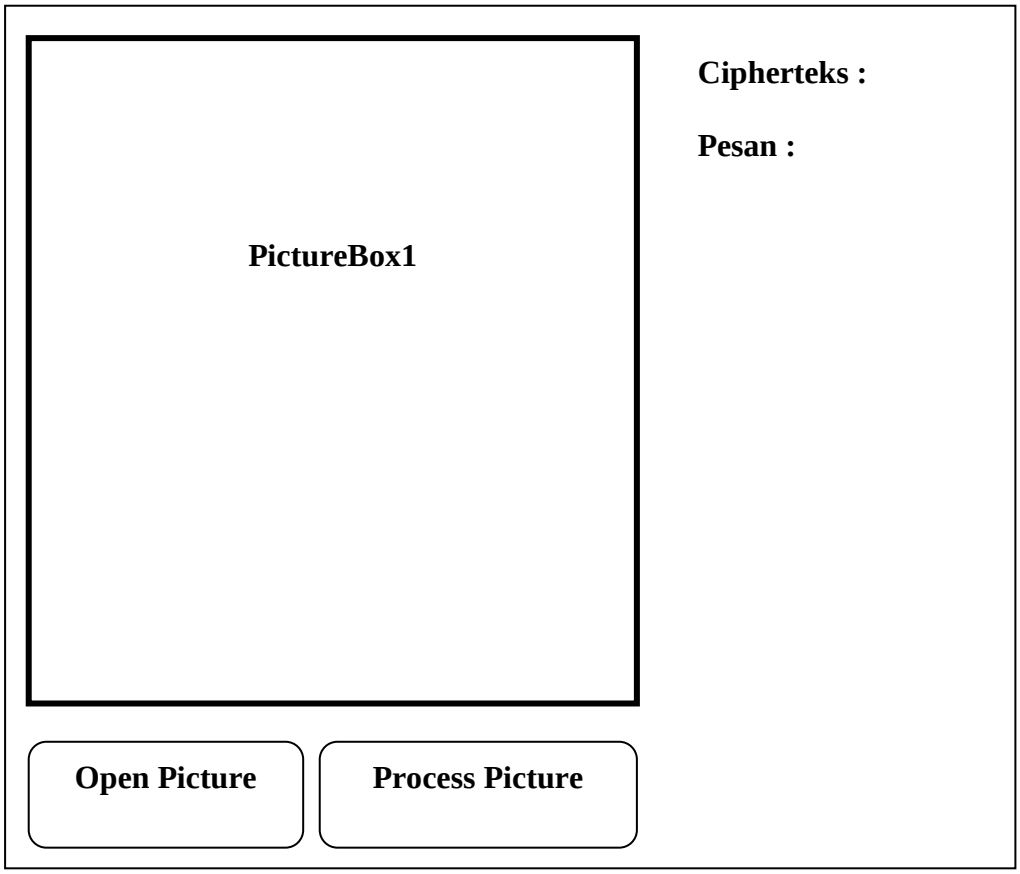
Gambar III.6 *Form* Sisip Pesan

3. Antarmuka *Form* Ekstrak Pesan

Rancangan *form* ini dibuat untuk melakukan proses pembacaan pesan dimana akan dijelaskan satu persatu tentang *form* tersebut. Berikut penjelasan form pembacaan pesan atau dapat dilihat pada gambar III.7 dibawah ini:

a. Picturebox untuk memunculkan gambar steganografi yang ingin dideskripsikan pesan.
b. Tombol/button open Picture untuk mengambil gambar steganografi yang ingin diproses.

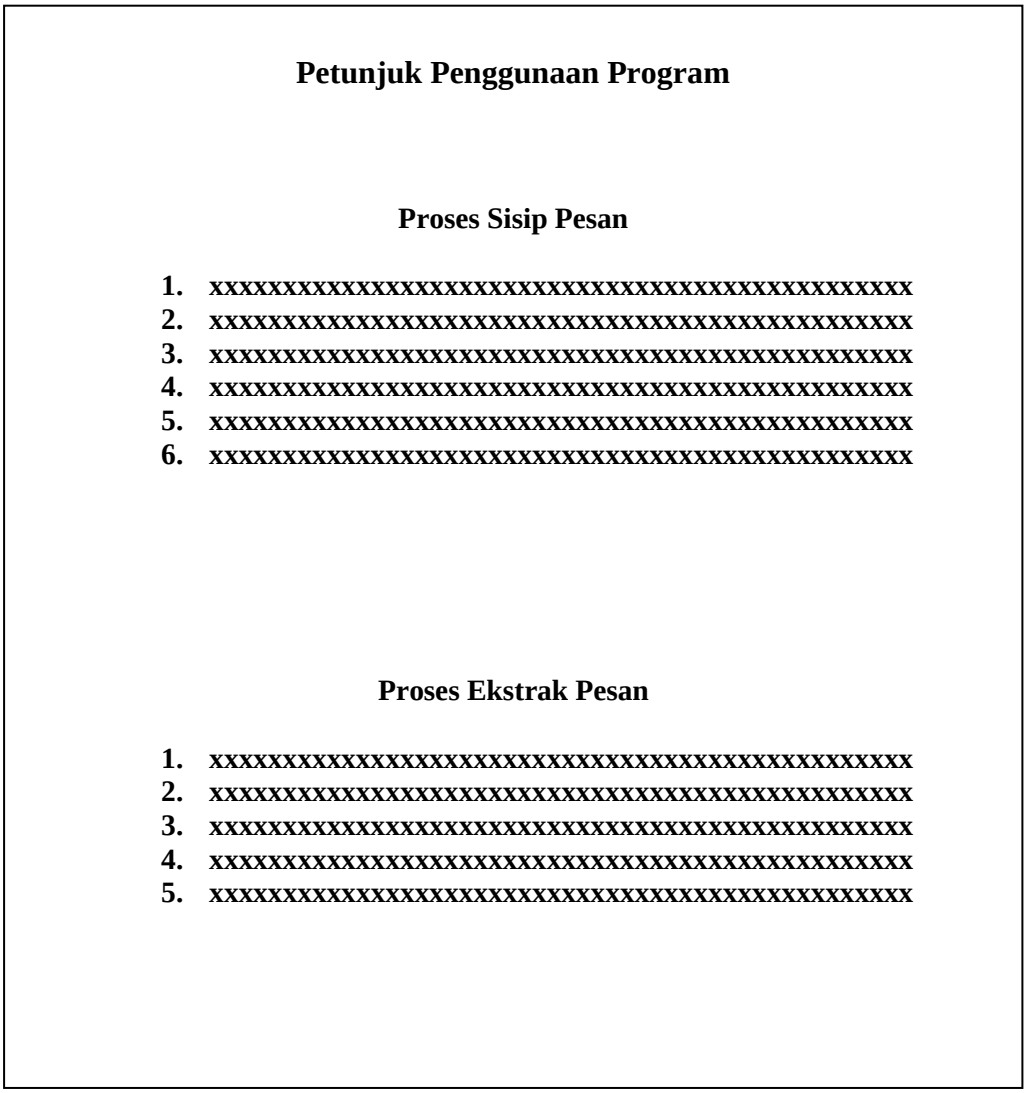
c. Tombol/button Process Picture merupakan tombol untuk menjalankan semua proses ekstrak gambar LSB dan deskripsi.



Gambar III.7 *Form* Ekstrak Pesan

4. Antarmuka *Form* Bantuan

Rancangan *form* ini dibuat untuk membantu *user/*pengguna dalam menggunakan program ini atau dapat dilihat pada gambar III.8 dibawah ini:



Gambar III.8 *Form* Bantuan