

BAB III

ANALISIS DAN DESAIN SISTEM

III.1. Analisis Masalah

Pesan terkadang mengandung sebuah informasi yang sangat penting yang harus dijaga kerahasiaannya. Ada beberapa cara yang dapat digunakan untuk menjaga kerahasiaan sebuah pesan, salah satunya adalah dengan menggunakan teknik steganografi dimana sebuah pesan disisipkan kedalam sebuah gambar dan secara visual tidak merubah tampilan gambar tersebut. Teknik steganografi sangat baik digunakan untuk menjaga kerahasiaan sebuah pesan karena orang lain tidak akan menyadari bahwa terdapat pesan rahasia pada sebuah gambar. Untuk itu dibutuhkan sebuah aplikasi yang dapat menyisipkan pesan kedalam sebuah gambar.

III.2. Steganografi LSB (*Least Significant Bit*)

Least significant bit adalah bagian dari barisan data *biner* (basis dua) yang mempunyai nilai paling tidak berarti/paling kecil. Letaknya adalah paling kanan dari barisan *bit*. Sedangkan *most significant bit* adalah sebaliknya, yaitu angka yang paling berarti/paling besar dan letaknya disebelah paling kiri.

Least significant bit sering kali digunakan untuk kepentingan penyisipan data ke dalam suatu media digital lain, salah satu yang memanfaatkan *Least significant bit* sebagai metode penyembunyian adalah steganografi *audio* dan gambar.

Metode yang digunakan untuk penyembunyian pesan rahasia pada aplikasi ini adalah dengan cara menyisipkan pesan ke dalam *bit* rendah (*least significant bit*) pada data *pixel* yang menyusun file gambar BMP 24 *bit* tersebut.

Pada file gambar BMP 24 *bit* setiap *pixel* pada gambar terdiri dari susunan tiga warna yaitu merah, hijau, biru (RGB) yang masing-masing disusun oleh bilangan 8 *bit* (1 *byte*) dari 0 sampai 255 atau dengan *format biner* 00000000 sampai 11111111. Sebagai contoh *file* gambar BMP 24 *bit* dengan warna merah murni dalam *format biner* akan terlihat sebagai berikut :

```
00000000 00000000 11111111
```

```
00000000 00000000 11111111
```

Sedangkan untuk warna hijau murni dalam *format biner* akan terlihat sebagai berikut :

```
00000000 11111111 00000000
```

```
00000000 11111111 00000000
```

Sedangkan untuk warna biru murni dalam *format biner* akan terlihat sebagai berikut :

```
11111111 00000000 00000000
```

```
11111111 00000000 00000000
```

Dari uraian di atas dapat dilihat bahwa informasi dari warna biru berada pada *bit* pertama sampai *bit* delapan, dan informasi warna hijau berada pada *bit* sembilan sampai dengan *bit* 16, sedangkan informasi warna merah berada pada *bit* 17 sampai dengan *bit* 24.

Metode penyisipan LSB (*least significant bit*) ini adalah menyisipi pesan

dengan cara mengganti *bit* ke 8, 16 dan 24 pada representasi *biner file* gambar dengan representasi *biner* pesan rahasia yang akan disembunyikan. Dengan demikian pada setiap *pixel file* gambar BMP 24 *bit* dapat disisipkan 3 *bit* pesan, misalnya terdapat data *raster original file* gambar adalah sebagai berikut :

00100111 11101001 11001000

00100111 11001000 11101001

11001000 00100111 11101001

Sedangkan representasi *biner* huruf A adalah 01000001, dengan menyisipkannya ke dalam *pixel* di atas maka akan dihasilkan :

0010011**0** 11101001 11001000

0010011**0** 11001000 1110100**0**

11001000 00100111 11101001

Terlihat pada *bit* kedelapan, enam belas dan 24 diganti dengan representasi biner huruf A, dan hanya tiga *bit* rendah yang berubah (cetak tebal), untuk penglihatan mata manusia sangatlah mustahil untuk dapat membedakan warna pada *file* gambar yang sudah diisi pesan rahasia jika dibandingkan dengan *file* gambar asli sebelum disisipi dengan pesan rahasia.

III.2.1. Metode LSB+1

Metode LSB+1 adalah Proses penyisipan pesan ke dalam citra digital dengan menggunakan metode LSB hampir sama saja dengan metode LSB, bedanya ada pada bit tempat penyisipan pesan. Jika pada metode LSB, pesan disisipkan pada bit LSB (bit ke-8), maka pada metode LSB+1, pesan disisipkan

pada bit ke-7. Sebagai contoh, misalkan tiga piksel yang berdekatan (sembilan bytes) dengan kode RGB berikut :

```
00110101    11010110    11101010
11110100    00111001    11100001
01110001    10010001    11100001
```

Pesan yang akan disisipkan adalah karakter “R”, yang nilai binernya adalah “01010010”. Pesan akan disisipkan dengan menggunakan metode LSB+1, maka akan dihasilkan citra hasil dengan urutan bit sebagai berikut:

```
00110101    11010110    11101000
11110110    00111001    11100001
01110011    10010001    11100001
```

Pada contoh di atas, dapat dilihat bahwa sebagian bit LSB+1 (bit ke-7) yang ada pada citra asal (original) digantikan dengan bit dari pesan yang akan disisipkan.

III.2.2. Metode LSB+2

Pada metode LSB+2, pesan disisipkan pada bit ke-6. Sebagai contoh, misalkan tiga piksel yang berdekatan dengan kode RGB seperti pada metode LSB+1 akan disisipkan adalah karakter “R” dengan menggunakan metode LSB+2, maka akan dihasilkan citra hasil dengan urutan bit sebagai berikut :

```
00110001    11010110    11101010
11110100    00111001    11100001
01110101    10010001    11100001
```

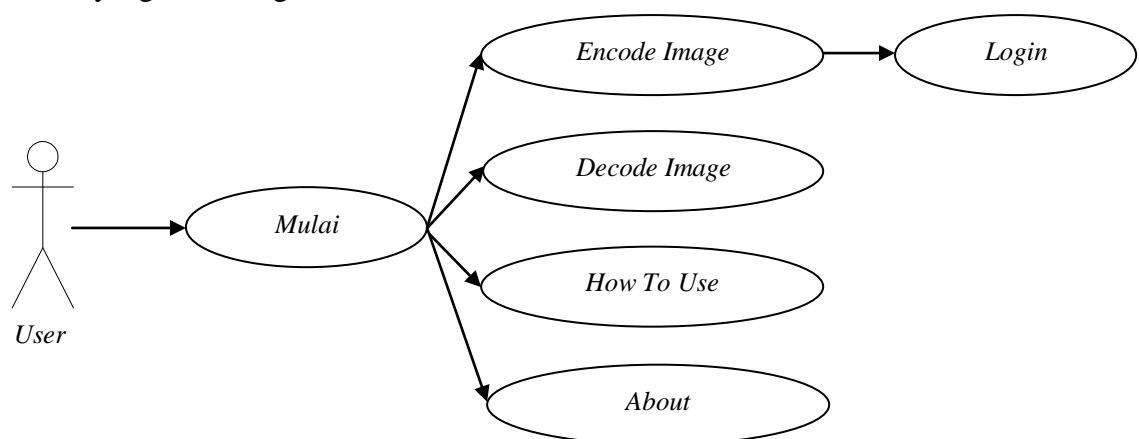
Pada contoh di atas, dapat dilihat bahwa sebagian bit LSB+2 (bit ke-6) yang ada pada citra asal (original) digantikan dengan bit dari pesan yang akan disisipkan.

III.3. Desain Sistem

Aplikasi Penyisipan Pesan Pada Citra Dengan Menggunakan Metode LSB+2 Berbasis Android dirancang dengan menggunakan perangkat lunak Eclipse Juno. Perancangan sistem yang dirancang terdiri dari *use case*, *flowchart*, *activity diagram* serta desain dan penjelasan dari sistem yang dirancang. Berikut adalah perancangannya :

III.3.1. Use Case Diagram

Use case mendiskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi yang ada didalam sistem informasi tersebut. Berikut adalah *use case diagram* dari sistem yang dirancang :

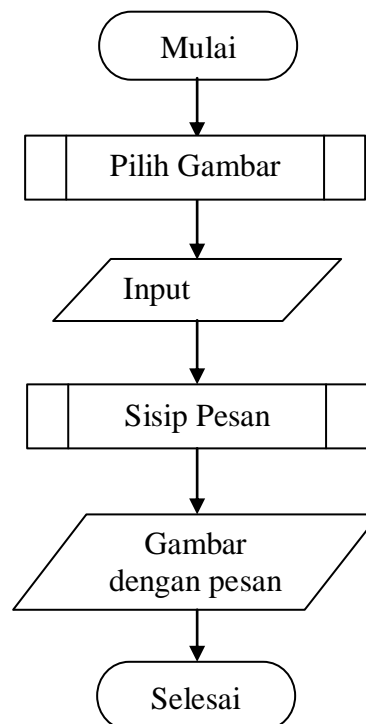


Gambar III.2. Use Case Diagram

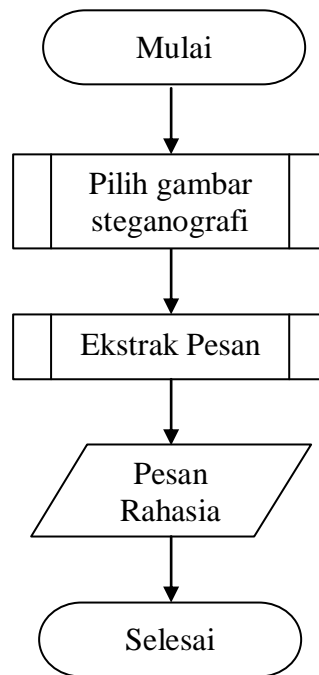
III.3.2. Flowchart

Flowchart adalah penggambaran secara grafik dari langkah-langkah dan urutan-urutan prosedur dari suatu program.

Tujuan utama dari penggunaan *flowchart* adalah untuk menggambarkan suatu tahapan penyelesaian masalah secara sederhana, terurai, rapi dan jelas dengan menggunakan simbol-simbol yang standar. Dalam perancangan aplikasi ini digunakan bagan alir (*flowchart*) untuk menjelaskan proses kerja dari perangkat lunak yang dirancang.



Gambar III.3. Flowchart Penyisipan Pesan



Gambar III.4. Flowchart Ekstrak Pesan

1. Proses *Flowchart* Penyisipan Pesan
 - a. Mulai
 - b. Pilih gambar
 - c. Masukkan pesan yang akan disisipkan
 - d. Lakukan proses steganografi
 - e. Gambar steganografi
 - f. Selesai
2. Proses *Flowchart* Ekstrak Pesan
 - a. Mulai
 - b. Pilih gambar steganografi
 - c. Ekstrak pesan menggunakan proses steganografi
 - d. Pesan rahasia ditampilkan

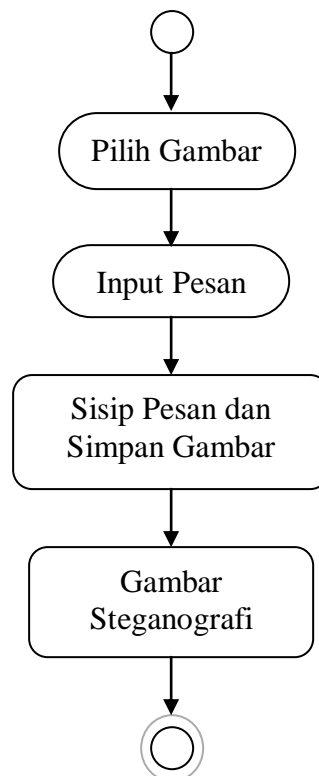
e. Selesai

III.3.3. Activity Diagram

Activity diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* yang terdapat pada aplikasi yaitu sebagai berikut :

1. Activity Diagram Penyisipan Pesan

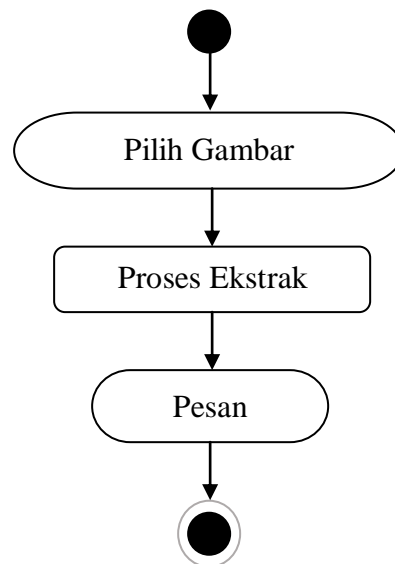
Activity diagram penyisipan pesan menggambarkan alir aktifitas penyisipan pesan yang dilakukan oleh pengguna dan diproses didalam sistem.



Gambar III.4 Activity Diagram Penyisipan Pesan

2. *Activity Diagram* Ekstrak Pesan

Activity diagram ekstrak pesan menggambarkan alir aktifitas ekstrak pesan yang dilakukan oleh pengguna dan diproses didalam sistem.



Gambar III.5 *Activity Diagram* Estrak Pesan

III.4. *Desain User Interface*

Antarmuka peamakai (*user interface*) adalah tampilan program yang dapat dilihat, didengar atau dipersepsikan oleh pengguna dan perintah-perintah atau mekanisme yang digunakan pemakai untuk mengendalikan operasi dan memasukkan data. Berikut ini merupakan perancangan aplikasi steganografi yang dirancang dengan antarmuka pada perangkat *mobile*, yaitu :

1. Desain *Form Login*

The diagram shows a rectangular form titled "Steganografi". Inside the form, there are three horizontal input fields stacked vertically. The top field is labeled "Username" and has a circled number "1" to its right. The middle field is labeled "Password" and has a circled number "2" to its right. The bottom field is labeled "::Login::" and has a circled number "3" to its right. Each field is connected to its respective number by a thin line.

Merupakan tampilan rancangan form login. Adapun keterangannya sebagai berikut :

- 1) *Textbox* yang digunakan untuk memasukkan *username*
- 2) *Textbox* yang digunakan untuk memasukkan *password*.
- 3) Tombol yang digunakan untuk masuk ke menu utama aplikasi.

2. Desain *Form Menu Utama*

The diagram shows a rectangular form titled "Steganografi". Below the title, there is a subtitle "Android Steganografi". Below the subtitle, there are four horizontal buttons stacked vertically. The top button is labeled "Encode Image" and has a circled number "1" to its right. The second button is labeled "Decode Image" and has a circled number "2" to its right. The third button is labeled "How To Use" and has a circled number "3" to its left. The bottom button is labeled "About" and has a circled number "4" to its right. Each button is connected to its respective number by a thin line.

Keterangan tampilan menu utama, yaitu :

- 1) Tombol untuk membuka halaman penyisipan pesan.
- 2) Tombol untuk membuka halaman ekstrak pesan.

- 3) Tombol untuk membuka halaman cara penggunaan aplikasi.
- 4) Tombol untuk membuka halaman tentang *programer*.

3. Desain *Form* Penyisipan Pesan

The diagram shows a form titled "Steganografi". It contains the following elements:

- A label "Gambar" and "Text to encode" with a text input field (callout 1).
- A "Pilih Gambar" button (callout 2).
- A large empty rectangular area (callout 3).
- A "Simpan" button (callout 4).

Tampilan form penyisipan pesan yang berfungsi sebagai tempat berlangsungnya proses penyembunyian pesan teks ke dalam gambar. Adapun keterangannya sebagai berikut :

- 1) Direktori dari gambar yang dipilih.
- 2) Tombol untuk memilih gambar.
- 3) *Textbox* untuk menginputkan pesan.
- 4) Tombol untuk menyisipkan pesan dan menyimpan gambar.

4. Desain *Form* Ekstrak Pesan

Steganografi
1

Merupakan tampilan dari *form* ekstrak pesan yang berfungsi sebagai proses pengambilan pesan yang tersisip didalam gambar. Adapun keterangan tampilan *form* ekstrak sabagai berikut :

- 1) *Textbox* untuk menampilkan pesan rahasia.

5. Desain *Form How To Use*

Steganografi
1

Adapun keterangannya sebagai berikut :

- 1) Menampilkan cara untuk melakukan penyisipan dan peng-ekstrakan pesan kedalam gambar.

6. Desain *Form About*

Steganografi
1

Adapun keterangannya sebagai berikut :

- 1) Menampilkan data dari *programer*.