

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Seringkali seseorang yang hendak mengirim pesan kepada orang lain, tidak ingin isi pesan tersebut diketahui oleh orang lain. Biasanya isi pesan tersebut bersifat sangat rahasia atau pribadi, yang hanya boleh diketahui antara pihak pengirim dan pihak penerima pesan, atau kalangan terbatas saja. Oleh karena itu, biasanya pengirim tersebut mengirim pesan secara tersembunyi agar tidak ada pihak lain yang mengetahui.

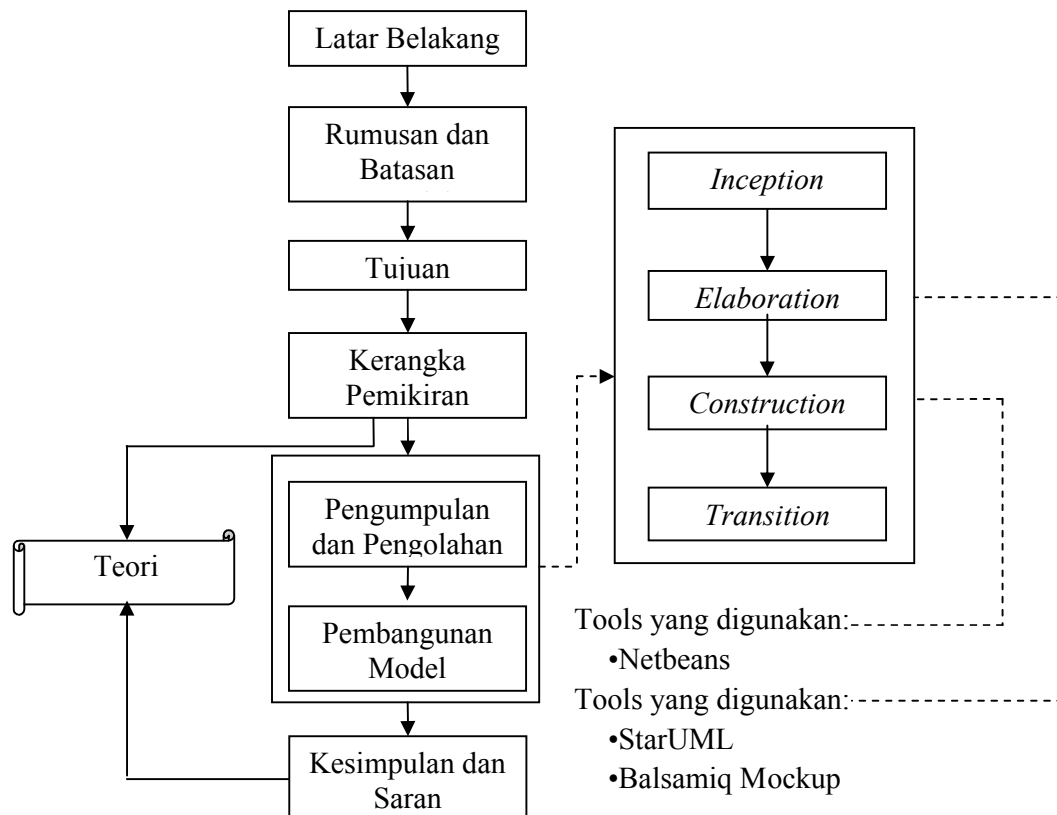
Salah satu hal yang dapat dilakukan untuk mengatasi situasi di atas adalah mengembangkan suatu aplikasi yang mampu menyamarkan pesan tersebut pada suatu media yang dapat diakses oleh setiap orang. Teknik ini disebut *steganography*, yaitu teknik penyembunyian data pada suatu media. Setiap orang bisa menampilkan atau membuka media tersebut, namun tidak menyadari bahwa media tersebut telah dibubuhkan pesan rahasia oleh pengirim. *Steganography* memungkinkan penyembunyian data pada berbagai jenis media digital seperti berkas citra, suara, video, dan teks.

Proses penyisipan pesan pada *steganography* membutuhkan dua buah masukan, yaitu pesan yang ingin disembunyikan, dan media penyisipan. Hasil dari proses ini dinamakan dengan *stego-object*, yaitu suatu media yang mirip dengan media pada masukan, yang sudah terdapat pesan tersembunyi di dalamnya. Kebanyakan media yang merupakan *stego-object* tidak dapat dikembalikan lagi seperti semula, karena data dari media *stego-object* sudah dirubah.

Teknik *steganography* yang digunakan adalah LSB (*Least Significant Bit*), metode ini banyak digunakan karena metode ini paling sederhana dan mudah diimplementasikan. Media penampung yang paling sering digunakan dalam melakukan *steganography* adalah citra digital karena jumlahnya yang besar di internet. Kehandalan penggunaan citra dibandingkan dengan media lain adalah kualitas citra yang telah disisipi pesan rahasia tidak berbeda jauh dengan kualitas citra aslinya.

## 1.2 Paradigma Penelitian

Paradigma penelitian dibuat untuk memperjelas langkah atau alur penelitian dengan menggunakan kerangka penelitian sebagai tahapan penelitian secara keseluruhan, dalam penelitian ini, paradigma penelitian digambarkan dalam bagan sebagai berikut:



**Gambar 1.1** Bagan Paradigma Penelitian

### 1.3 Rumusan Masalah

Sesuai dengan latar belakang yang telah dijelaskan di atas, maka dapat dirumuskan sebuah pokok permasalahan, yaitu:

- 1) Bagaimana mengimplementasikan *steganography* sebagai salah satu teknik untuk mengamankan pesan rahasia dalam citra digital berformat *PNG* dengan metode *LSB (Least Significant Bit)*?
- 2) Bagaimana menganalisis perubahan *file* citra hasil keluaran baik itu kualitas *file* maupun besar data *file* dan seberapa besar perubahan itu terjadi setelah dilakukan proses *encode* dan *decode* penyisipan pesan rahasia?

### 1.4 Batasan Masalah

Dalam skripsi ini pembahasan hanya terbatas pada pengamanan data untuk data teks dengan media citra berformat *PNG* dengan menerapkan metode *LSB (Least Significant Bit)*. Sedangkan pengujian *Robustness* tidak dilakukan karena yang utama steganografi bertujuan untuk menghindari kecurigaan (lawan tidak menyadari keberadaan pesan tersembunyi).

### 1.5 Tujuan Penulisan Skripsi

Tujuan Tujuan dari penulisan skripsi ini adalah :

- 1) Mengimplementasikan teknik *steganography* dengan metode *LSB (Least Significant Bit)* untuk mengamankan pesan rahasia dalam citra digital berformat *PNG*.
- 2) Melakukan analisis perubahan *file* citra hasil keluaran baik itu kualitas *file* maupun besar data *file* dan seberapa besar perubahan itu terjadi setelah dilakukan proses *encode* dan *decode* penyisipan pesan rahasia.

- 3) Melakukan analisis terhadap citra stego setelah dilakukan penambahan operasi manipulasi.

## **1.6 State of The Art**

Teknik dalam melakukan steganografi ada bermacam-macam. Setiap metode memiliki karakteristik masing-masing. Berikut akan dibahas metode pembandingan untuk metode *Least Significant Bit*, yaitu metode *GIFShuffle* dan *Spread Spectrum*.

### **1.6.1 GIFShuffle**

Metode Gifshuffle pada intinya memanfaatkan palet warna berkas GIF sebagai media penyisipan pesan. Dalam metode ini tidak terjadi perubahan apapun dalam data berkas GIF. Sehingga menambah aspek robustness dari metode ini.

Sesuai dengan namanya Gifshuffle akan melakukan “*shuffle*” palet warna dari sebuah berkas GIF *shuffle* jika diterjemahkan ke dalam bahasa GIF, Indonesia berarti mengacak. Sehingga dapat diartikan bahwa GifShuffle adalah metode yang memanfaatkan penukaran posisi ke 256 palet warna dalam berkas palet citra berformat GIF. Dalam 256 palet warna tersebut kita dapat menyimpan  $\log_2(256!) \text{ bit} = 1675 \text{ bit} (209 \text{ bytes})$  pesan berdasarkan pengurutannya. Hal tersebut aman dilakukan karena dua buah berkas GIF dengan palet warna yang berbeda akan ditampilkan secara sama persis.

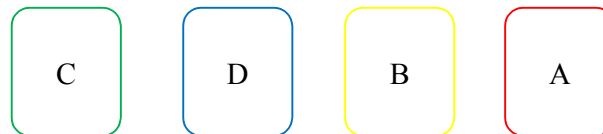
Dengan dilakukannya penukaran posisi maka akan dapat diperoleh sebuah informasi berkaitan dengan perbedaan posisi akhir dengan posisi awal. Dengan kata lain jika kita diberikan  $n$  sebagai ukuran media penyisipan maka kita dapat menyimpan  $\log_2(n!)$  bit informasi berdasarkan pengurutannya.

Sebagai contoh adalah jika kita mengurutkan 4 kartu menurut besarnya sebagai berikut :



**Gambar 1.2** Posisi Kartu Awal

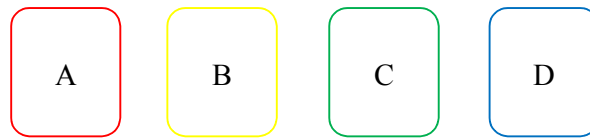
Kemudian karena kita ingin menyimpan informasi maka kita dapat mengganti pengurutannya sebagai berikut :



**Gambar 1.3** Posisi Kartu Akhir

Pergantian urutan dari sebuah kartu dapat diasosiasikan dengan informasi. Sebagai contoh dalam ilustrasi diatas. Jika kartu A bergeser 3 kali ke kiri maka dapat diartikan bahwa informasi yang terkandung dalam pergeseran tersebut adalah huruf a. Atau jika kartu B bergeser sejauh 1 kartu dapat diartikan huruf c. Hal tersebut tergantung kepada kesepakatan di awal. Ilustrasi diatas adalah sedikit ilustrasi untuk menggambarkan bagaimana cara metode GifShuffle menyisipkan pesan dalam sebuah berkas GIF yaitu dengan mengganti susunan palet warna dan mengasosiasikan sebuah informasi berkaitan dengan pergeseran tersebut (Agustinus, 2005).

Contoh lainnya, ada sebuah pesan yang ingin disisipkan, diasumsikan bilangan dari hasil konversi desimal pesan tersebut adalah 22, sehingga M adalah konversi desimal pesan. Diasumsikan media penyisipan pesan berukuran 4, sehingga N adalah ukuran media penyisipan. Maka susunan awal media penyisipan adalah sebagai berikut :



**Gambar 1.4** Media Penyisipan

Kemudian dilakukan iterasi terhadap variabe  $i$  dengan nilai  $i$  dari 1 sampai  $N$ .  
Setiap palet warna dengan urutan  $N$  dipindahkan ke posisi baru yaitu  $M \bmod i$ , kemudian  $M$  dibagi dengan  $i$ .

1. Palet D

$$22 \bmod 1 = 0$$

$$M = 22 \text{ div } 1$$

$$M = 22$$

Sehingga palet D menempati posisi ke-0 pada susunan yang baru.

2. Palet C

$$22 \bmod 2 = 0$$

$$M = 22 \text{ div } 2$$

$$M = 11$$

Sehingga palet C menempati posisi ke-0 pada susunan yang baru.

3. Palet B

$$11 \bmod 3 = 2$$

$$M = 11 \text{ div } 3$$

$$M=3$$

Sehingga palet B menempati posisi ke-2 pada susunan yang baru.

4. Palet A

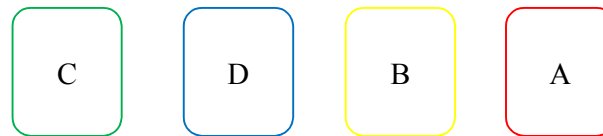
$$3 \bmod 4 = 3$$

$$M = 3 \text{ div } 4$$

$$M=0$$

Sehingga palet A menempati posisi ke-3 pada susunan yang baru.

Kemudian palet warna yang baru hasil iterasi pada langkah diatas dimasukkan ke dalam palet warna berkas GIF. Apabila ada sebuah tempat yang diisi oleh 2 buah warna maka warna yang sebelumnya menempati tempat tersebut akan digeser satu tempat ke setelahnya. Karena huruf C dan D menempati posisi yang sama yaitu posisi ke-0. Sehingga sekarang posisi palet D digeser 1 blok ke kanan untuk menempati posisi ke-1. Hasil akhir dari penggantian susunan palet-palet ini adalah :



**Gambar 1.5** Media Yang Telah Disisipkan

Kelemahan dari algoritma Gifshuffle adalah sangat terbatasnya panjang pesan yang dapat disisipkan karena jumlah warna pada palet warna citra GIF adalah maksimal 256 warna.

### 1.6.2 Spread Spectrum

*Spread spectrum* merupakan bagian dari teknik dalam ranah *transform*. *Spread spectrum* merupakan teknik penransmisian dengan menggunakan *pseudo-noise code*, yang independen terhadap data informasi, sebagai modulator bentuk gelombang untuk menyebarkan energi sinyal dalam sebuah jalur komunikasi (*bandwidth*) yang lebih besar

daripada sinyal jalur komunikasi informasi. Oleh penerima, sinyal dikumpulkan kembali menggunakan replika *pseudo-noise code* tersinkronisasi.

Berdasarkan definisi, dapat dikatakan bahwa steganografi menggunakan metode *spread spectrum* memperlakukan *cover-object* baik sebagai derau (*noise*) ataupun sebagai usaha untuk menambahkan derau semu (*pseudonoise*) ke dalam *cover-object*.

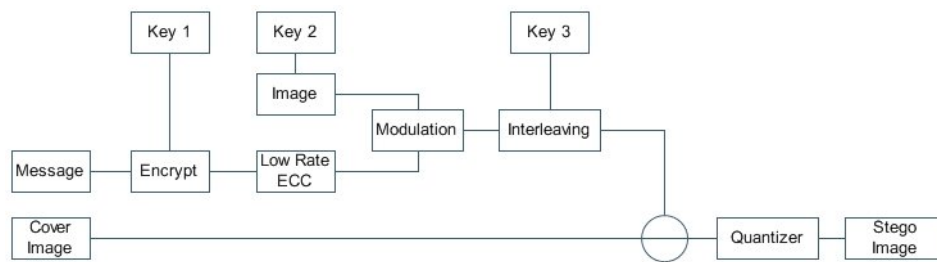
Sistem yang memperlakukan *cover-object* sebagai derau dapat menambahkan sebuah nilai ke dalam *cover-object*. Nilai ini harus ditransmisikan di bawah tingkat derau yang ditambahkan nilai ke dalamnya. Hal ini berarti kapasitas sangat ditentukan oleh *cover-object*. Sementara nilai yang disisipkan dapat berupa bilangan real, dalam prakteknya, sulit untuk memasukkan dan mengekstraksi nilai real dari satu bit data. Untuk dapat melakukan transmisi lebih dari satu bit, *cover-object* dibagi menjadi bagian-bagian kecil, disebut dengan *sub-cover-object*. Ketika *sub-cover-object* rata (*tile*) metode ini dikatakan menggunakan *direct sequence spread spectrum steganography*. Ketika *sub-cover-object* terdiri dari titik-titik terpisah yang terdistribusi di seluruh *cover-object*, dalam hal ini *cover-image*, metode ini disebut sebagai *frequency-hopping spread spectrum steganography*. Metode ini memerlukan proses pencarian secara menyeluruh terhadap *cover-object* untuk mendapatkan *carrier* (yang akan disisipi informasi) dan memaksimalkan penggunaan data yang terkandung di dalam *cover-object*.

Penggunaan *direct-sequence* memerlukan pemrosesan yang jauh lebih ringan dibandingkan dengan *frequency-hopping*. Di sisi lain, *direct-sequence* tidak dapat mengatasi *cropping*, sementara *frequency-hopping* hanya akan menganggap hal tersebut sebagai sedikit pengurangan energi sinyal. Kapasitas dapat dipertukarkan secara linear dengan kekokohan. Hal ini sebenarnya sangat bergantung pada *cover-object* yang digunakan. Penyaringan spasial dapat menghilangkan informasi dalam *stego-object* jika



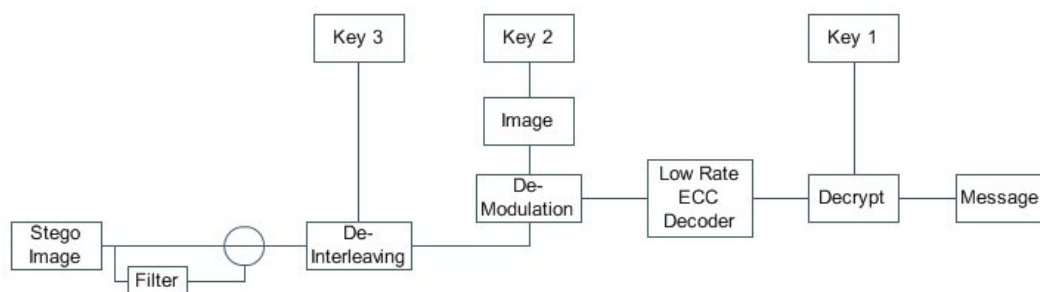
penggunaan interval cenderung sama dalam setiap *sub-cover-object*. Interval yang acak dapat membantu mengatasi hal ini secara signifikan.

Metode yang menempatkan informasi di dalam derau semu di keseluruhan *cover-image* disebut sebagai *spread spectrum image steganography*. Penggunaan metode ini di deskripsikan dalam Gambar 1.6 dan Gambar 1.7 masing-masing menunjukkan langkahlangkah dalam melakukan penyisipan dan pengekstrasian informasi ke dan dari dalam *cover-image*.



**Gambar 1.6** Penyisipan Informasi Menggunakan Metode Spread Spectrum Image Steganography

(Sumber: *Spread Spectrum Image Steganography*.1999)



**Gambar 1.7** Pengekstrasian informasi menggunakan metode spread spectrum image steganography

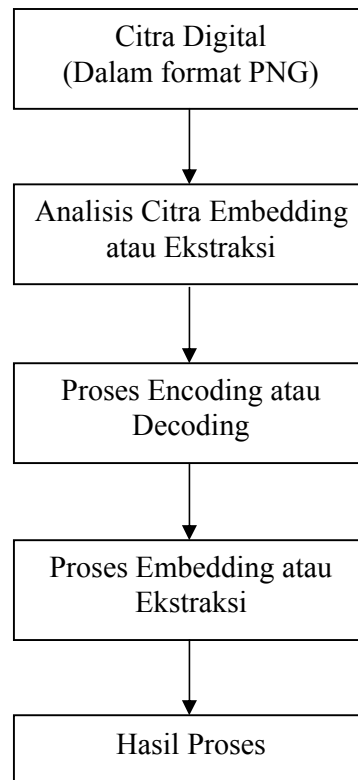
(Sumber: *Spread Spectrum Image Steganography*.1999)

Untuk melakukan penyisipan, informasi diubah terlebih dahulu menjadi derau semu yang kemudian dimasukkan ke dalam *coverimage* untuk menghasilkan *stego-image*. Untuk melakukan pengekstrasian, *stego-image* disaring untuk mendapatkan derau semu dan kemudian derau semu tadi diekstraksi menjadi informasi. Ada tiga kunci yang digunakan dalam metode ini. Kunci yang pertama digunakan untuk mengenkripsi informasi. Kunci yang kedua digunakan dalam membangkitkan derau semu. Dan, kunci yang ketiga digunakan untuk memasukkan dan menyebarkan informasi yang telah dimodulasi ke dalam *cover-image* sedemikian rupa sehingga penyisipan informasi sesedikit mungkin mempengaruhi *cover-image*.

Kelemahan metode ini adalah lamanya proses *embedding* dan *ekstraksi* dikarenakan proses modulasi antara pesan dan kata kunci.

### **1.7 Kerangka Pemikiran**

Kerangka pemikiran merupakan urutan logis proses untuk dapat memecahkan suatu masalah penelitian. Gambar 1.8 menunjukkan kerangka pemikiran aplikasi yang akan dibangun. Langkah pertama yaitu membuka *file* citra. Format citra digital yang dapat diproses adalah PNG. Setelah itu dilakukan analisa citra yang terbagi atas dua proses yaitu *embedding* dan ekstraksi. Pada analisa citra *embedding*, citra induk harus lebih besar dari citra anak dan pada ekstraksi akan dianalisa bahwa *stegoimage* yang dibuka adalah hasil dari *embedding* sebelumnya. Kemudian dilakukanlah proses *embedding* atau ekstraksi yang akan menghasilkan *stegoimage* dari proses *embedding* dan *hiddenimage* dari proses ekstraksi.



**Gambar 1.8** Kerangka Pemikiran

### ***Least Significant Bit***

*Bit* atau *binary digit* adalah unit dasar penyimpanan data di dalam komputer, nilai *bit* suatu data adalah 0 atau 1. Semua data yang ada pada komputer disimpan ke dalam satuan bit ini, termasuk gambar, suara, ataupun video. Jenis-jenis format pewarnaan di dalam media gambar, seperti *grayscale*, RGB, dan CMY. Sebagai contoh pewarnaan monochrome bitmap (menggunakan 1 bit untuk tiap pixelnya), RGB - 24 bit (8 bit untuk Red, 8 bit untuk Green, dan 8 bit untuk Blue), Grayscale-8 bit (menentukan tingkat kehitaman suatu pixel berdasarkan nilai bitnya).

Misalkan sebuah data berupa text “secret“, kalau direpresentasikan ke dalam binary kata “secret“ ini menjadi:

**Tabel 1.1** Tabel Representasi Kata “secret”

Character	ASCII value (decimal)	Hexadecimal	Binary
s	115	73	01110011
e	101	65	01100101
c	99	63	01100011
r	114	72	01110010
e	99	63	01100011
t	116	74	01110100

Sesuai dengan namanya, LSB artinya *bit* yang tidak *significant* / tidak mempunyai pengaruh yang besar, maka metode ini mengganti nilai *bit* ke-8 sebuah gambar untuk menyisipkan data. Berikut permisalan data *binary* dari media gambar tersebut:

**Tabel 1.2.** Data *Binary* Media

00000000	00000000	00000001	00000001	00000001	00000001	00000001	00000001
00000000	00000000	00000001	00000001	00000001	00000001	00000001	00000001
00000000	00000000	00000001	00000001	00000001	00000001	00000001	00000001
00000001	00000001	00000010	00000010	00000010	00000011	00000011	00000011
00000001	00000001	00000010	00000010	00000010	00000011	00000011	00000011
00000001	00000001	00000010	00000010	00000010	00000011	00000011	00000011

Data *binary* yang akan disisipkan seperti berikut:

**Tabel 1.3.** Tabel *Binary* Data Yang Akan Disisipkan

0	1	1	1	0	0	1	1
0	1	1	0	0	1	0	1
0	1	1	0	0	0	1	1
0	1	1	1	0	0	1	0
0	1	1	0	0	0	1	1
0	1	1	1	0	1	0	0

Hasil akhir data *binary* dari media gambar yang telah disisipkan sebagai berikut:

**Tabel 1.4.** Tabel Hasil Berkas *Stego*

00000000	00000001	00000001	00000001	00000000	00000000	00000001	00000001
00000000	00000001	00000001	00000000	00000000	00000001	00000000	00000001
00000000	00000001	00000001	00000000	00000000	00000000	00000001	00000001
00000000	00000001	00000011	00000011	00000010	00000010	00000011	00000010
00000000	00000001	00000011	00000010	00000010	00000010	00000011	00000011
00000000	00000001	00000011	00000011	00000010	00000011	00000010	00000010

Angka yang di-*bold* menunjukkan kalau data tersebut sudah diganti sesuai dengan data yang ingin disisipkan. Tabel-tabel diatas telah menjelaskan bagaimana cara metode LSB (*Least Significant Bit*) bekerja.

## 1.8 Metode Penyusunan Skripsi

Metodologi penelitian yang digunakan dalam penyusunan skripsi ini adalah sebagai berikut :

### 1) Tahap Pengumpulan Data

#### a. Eksplorasi dan Studi Literatur

Eksplorasi dan studi literatur dilakukan dengan mempelajari konsep-konsep yang berkaitan dengan skripsi ini, seperti *steganography*, metode LSB (*Least Significant Bit*), citra digital, melalui literatur-literatur seperti buku (*textbook*), paper, dan sumber ilmiah lain seperti situs internet ataupun artikel dokumen teks yang berhubungan.

#### b. Observasi

Teknik pengumpulan data dengan mengadakan penelitian dan peninjauan langsung terhadap permasalahan yang diambil.

## 2) Tahap Pembuatan

Teknik analisis data dalam pembuatan perangkat lunak ini menggunakan metode pengembangan RUP (*Rational Unified Process*), yang meliputi beberapa fase diantaranya:

### a. Tahap Insepsi

Tahap insepsi adalah tahap persiapan. Insepsi memiliki tujuan untuk menentukan manfaat dari perangkat lunak yang akan dibangun, dan perencanaan dari proyek yang terkait dengan latar belakang masalah, tujuan organisasi, proses bisnis yang ada, masalah serta solusi.

### b. Tahap Elaborasi

Tahap elaborasi merupakan tahap perencanaan di mana penekanan dilakukan pada terselesaikannya deskripsi kebutuhan perangkat lunak (termasuk survei), analisis dan desain arsitektur, serta pembangunan kerangka dasar sistem dan metode pengujiannya sehingga dapat memberi dasar/patokan untuk fase selanjutnya(konstruksi). Baseline arsitektur aplikasi berupa dokumen, use case diagram, classdiagram, sequence diagram, dan activity diagram.

### c. Tahap Konstruksi

Konstruksi merupakan tahap ketiga pada pengimplementasian pengembangan aplikasi ini. Pada tahap sebelumnya telah dilakukan penentuan terhadap use case dan rancangan arsitektur awal, sehingga pada fase konstruksi ini akan melakukan penggambaran story board dan implementasi kode program sesuai dengan perancangan sebelumnya yakni use case dan diagram lainnya serta melakukan testing terhadap aplikasi yang dikembangkan.

d. Tahap Transisi

Pada fase transisi perangkat lunak yang dibangun telah selesai. Pada tahap ini dilakukan pengujian terhadap aplikasi yang telah dibangun kepada pengguna dan user acceptance test (dilampirkan), untuk melihat sistem yang ada sudah tidak ada masalah lagi pada saat pengujian dan sudah sesuai dengan keinginan pengguna.

### **1.9 Sistematika Penulisan Skripsi**

Sistematika penulisan Skripsi ini adalah sebagai berikut:

a. BAB I: PENDAHULUAN

Menguraikan tentang latar belakang permasalahan, mencoba merumuskan inti permasalahan yang dihadapi, batasan masalah, tujuan penulisan skripsi, yang kemudian diikuti dengan kerangka pemikiran, lalu metode penelitian serta sistematika penulisan.

b. BAB II: LANDASAN TEORI

Melampirkan berbagai konsep dasar dan teori-teori yang berkaitan dengan tahap penelitian yang dilakukan dan hal-hal yang berguna dalam proses analisis permasalahan.

c. BAB III: ANALISIS DAN PERANCANGAN

Menganalisis masalah serta merancang sistem yang akan diimplementasikan pada tahap berikutnya.

d. BAB IV: IMPLEMENTASI DAN PENGUJIAN

Merupakan tahapan yang dilakukan dalam penelitian secara garis besar sejak dari tahap persiapan sampai penarikan kesimpulan, metode dan kaidah yang diterapkan dalam penelitian. Termasuk menentukan variable penelitian, identifikasi data yang diperlukan dan cara pengumpulannya, penentuan sampel penelitian dan teknik

pengambilannya, serta metode/teknik analisis yang akan dipergunakan dan perangkat lunak yang akan dibangun. Serta melakukan tahap pengujian setelah implementasi selesai.

e. **BAB V: KESIMPULAN DAN SARAN**

Berisi kesimpulan dan saran yang sudah diperoleh dari hasil penulisan skripsi.