

## **BAB III**

### **METODOLOGI PENELITIAN**

#### **3.1 Alat dan Bahan Penelitian**

##### **3.1.1 Alat Penelitian**

Alat yang digunakan dalam penelitian yaitu dari segi perangkat keras dan perangkat lunak, yaitu :

##### 1. Perangkat Keras

- Processor Intel Core 2 Duo(tm) 6300, 1,8 GHz
- Ram 1 GB DDR2
- VGA Ge Force HD 9400
- Hardisk 120 GB

##### 2. Perangkat Lunak

- Borland Delphi 7
- ACD See 10

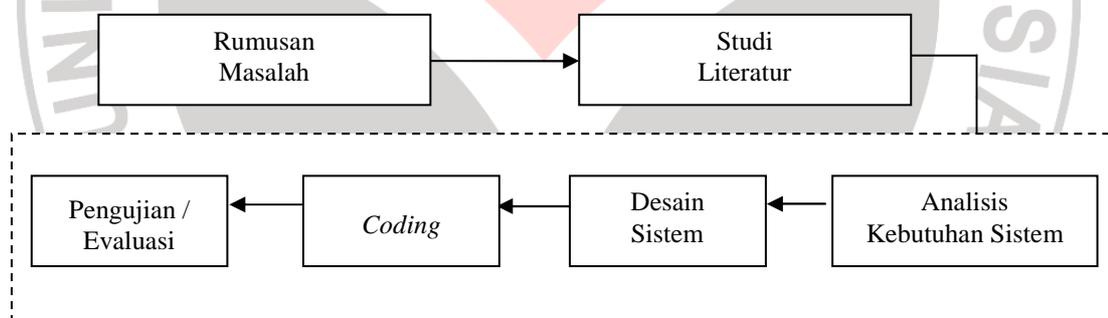
##### **3.1.2 Bahan Penelitian**

Bahan penelitian merupakan objek dari penelitian ini yang akan diolah oleh sistem, sistem yang dimaksud adalah aplikasi penyembunyian pesan pada citra digital. Objek yang dijadikan penelitian yaitu berupa citra penampung pesan. Citra yang dipakai adalah berkas citra dengan format GIF 8 bit non-transparan. Pada GIF animasi tidak ada standard bagaimana harus ditampilkan sehingga umumnya *image viewer* hanya akan menampilkan citra pertama dari berkas GIF,

bila berkas GIF animasi terjadi perubahan maka tidak semua bagian dalam GIF animasi ditampilkan kembali, hanya bagian yang berubah saja yang ditampilkan. Untuk itu pada penelitian ini digunakan citra GIF non-animasi (FTI Univ. Kristen Duta Wacana, 2006). Seperti kita ketahui untuk standard berkas GIF hanya menggunakan 256 palet warna yang terdiri dari susunan tiga warna yaitu merah, hijau, biru ( $R_0G_0B_0R_1G_1B_1 \dots R_{255}G_{255}B_{255}$ ). Metode yang digunakan dalam perangkat lunak ini adalah metode Gifshuffle.

### 3.2 Desain Penelitian

Desain penelitian yang digunakan dalam penelitian ini dijelaskan dalam gambar di bawah ini :



**Gambar 3.1** *Desain Penelitian*

Keterangan dari gambar di atas, yaitu :

#### 1. Rumusan Masalah

Perumusan masalah dalam penelitian ini dapat dilihat pada bagian pendahuluan dalam perumusan masalah.

## 2. Studi Literatur

Proses studi literatur dalam penelitian ini dilakukan dengan mempelajari literatur-literatur yang meliputi konsep steganografi, format citra *GIF* dan *algoritma gifshuffle*.

## 3. Analisis Kebutuhan Sistem

Hasil dari proses pengumpulan data yang telah dilakukan selanjutnya dianalisis dan dilakukan perumusan kebutuhan dari perangkat lunak yang akan dibuat.

## 4. Desain Sistem

Setelah dilakukan analisis sistem maka kebutuhan perangkat lunak dituangkan kedalam sebuah model perangkat lunak yang meliputi pemodelan data.

## 5. Coding

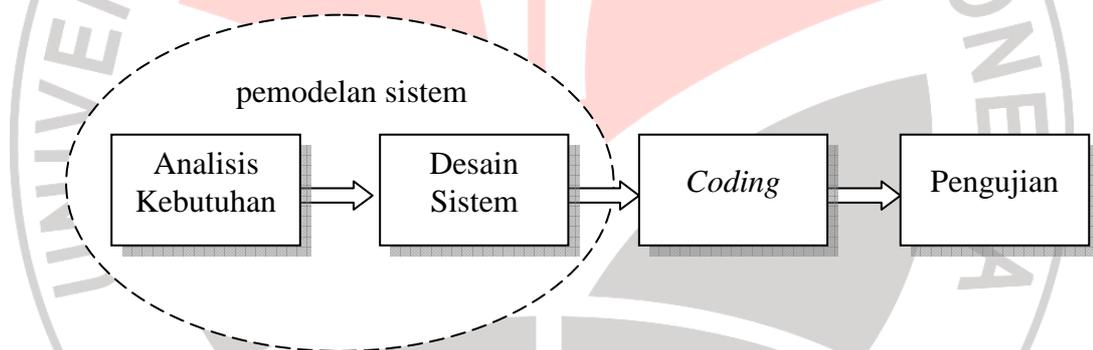
Setelah dilakukan desain sistem maka proses selanjutnya adalah proses *coding*, dalam penelitian ini perangkat lunak yang dibuat menggunakan bahasa pemrograman Borland Delphi 7.

## 6. Pengujian/Evaluasi

Tahap selanjutnya dilakukan proses pengujian, teknik pengujian yang digunakan adalah teknik pengujian *Black Box*.

### 3.3 Metode Pengembangan Perangkat Lunak

Metode pengembangan perangkat lunak merupakan tahapan pemodelan rekayasa perangkat lunak pada penelitian. Dalam penelitian ini, metode pengembangan perangkat lunak menggunakan *linear sequential model*, atau sering juga disebut dengan *classic life cycle* atau *waterfall model*. Model ini adalah model yang muncul sekitar tahun 1970 sehingga sering dianggap kuno, tetapi merupakan model yang paling banyak dipakai di dalam *software engineering* (SE). Adapun tahapan pengembangan dengan *waterfall model* ini menurut pressman adalah :



**Gambar 3. 2** model *waterfall* Pengembangan Perangkat Lunak

Keterangan dari gambar model *waterfall* diatas, yaitu :

1. Pada tahap analisis kebutuhan Perangkat lunak merupakan tahapan awal dalam analisis kebutuhan sistem. Tujuan dari tahap ini yaitu untuk mengetahui informasi, model, dan spesifikasi dari sistem yang dibutuhkan. Untuk mewujudkan tujuan dari proses ini maka penulis melibatkan pihak rumah sakit yang akan menggunakan sistem ini.

2. Setelah proses analisis kebutuhan selesai dilakukan, selanjutnya hasil analisis tersebut akan dimodelkan, model yang dibangun merujuk pada pendekatan pengembangan perangkat lunak berbasis aliran. Ada empat atribut yang dijadikan pada tahapannya yaitu struktur data, arsitektur perangkat lunak, representasi *interface* dan detail algoritma.
3. Setelah dilakukan tahap desain sistem maka proses selanjutnya adalah proses coding. Proses *coding* ini menterjemahkan desain yang telah dibuat kedalam kode-kode dengan menggunakan bahasa pemrograman. Penelitian ini menggunakan bahasa pemrograman Borland Delphi 7.
4. Tahapan selanjutnya adalah proses pengujian perangkat lunak, proses pengujian ini dilakukan untuk memastikan perangkat lunak yang telah dibuat telah sesuai dengan kebutuhan. Semua tahapan beserta detail penjelasan dari pembangunan perangkat lunak ini dituangkan dalam dokumen SKPL (Spesifikasi Kebutuhan Perangkat Lunak).

### **3.4 Implementasi**

Dibawah ini akan dijelaskan konsep dari metode yang akan diterapkan pada system.

#### **3.1.1 Konsep Steganografi**

Penilaian kinerja algoritma steganografi harus memenuhi empat faktor yaitu *imperceptible*, *fidelity*, *recovery* dan *robustnes* . Adapun langkah-langkah kegiatan pengimplementasian sistem ini adalah :

1. Pemahaman Konsep Steganografi pada Citra Digital.
2. Menerapkan algoritma *Gifshuffle* dengan menyisipkan pesan dengan memanfaatkan penukaran posisi ke 256 palet warna dalam berkas citra berformat GIF.
3. Membuat sistem steganografi pada citra digital berformat GIF yaitu menerjemahkan hasil rancangan sistem sehingga nantinya rancangan sistem tersebut dapat diproses oleh komputer.

### 3.1.2 Konsep Algoritma *Gifshuffle*

Salah satu mekanisme untuk menyelesaikan masalah steganografi pada citra digital adalah dengan mengaplikasikan algoritma *gifshuffle*. Algoritma *gifshuffle* adalah algoritma yang menyisipkan pesan dengan cara melakukan “*shuffle*” terhadap palet warna dari sebuah berkas GIF, *shuffle* jika diterjemahkan ke dalam bahasa Indonesia berarti mengacak. Sehingga dapat diartikan bahwa *GifShuffle* adalah algoritma yang memanfaatkan penukaran posisi ke 256 palet warna dalam berkas citra berformat GIF.

Pengimplementasian algoritma *gifshufle* ini ke dalam aplikasi steganografi dilakukan dengan langkah-langkah sebagai berikut :

1. Tentukan berkas digital yang akan dijadikan media penampung.
2. Mengambil nilai palet berdasarkan posisi indeks palet citra tersebut.
3. Konversi setiap warna dengan format RGB ke bilangan integer dengan aturan (merah\* 65536 + hijau \* 256 + biru ). Kemudian diurutkan berdasarkan besar bilangan interger yang mewakili warna tersebut dari kecil ke besar. Warna yang sama dihapus.

4. Masukkan pesan yang akan disisipkan.
5. Asumsikan kumpulan representasi biner yang tadi diperoleh sebagai sebuah angka. Biasanya langkah ini akan menghasilkan sebuah bilangan yang sangat besar karena konversi diperoleh dari biner yang besar. Namakan bilangan yang diperoleh ini sebagai  $M$ .
6. Lakukan iterasi terhadap variabel  $i$  dengan nilai  $i$  dari 1 sampai  $N$ . Setiap warna dengan urutan  $N-i$  dipindahkan ke posisi baru yaitu  $M \bmod i$ , kemudian  $M$  dibagi dengan  $i$ .
7. Masukkan palet warna baru hasil iterasi pada langkah 6 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 berikutnya.
8. Apabila besar dari palet warna yang baru lebih kecil dari 256, maka palet warna berikutnya akan diisi dengan warna terakhir dari palet warna sebelumnya.
9. Kemudian palet warna yang baru pada berkas GIF menghasilkan berkas GIF yang baru dengan ukuran dan citra yang sama namun telah disisipi pesan (D. Ariyus, 2009)