

BAB III

ANALISIS MASALAH DAN RANCANGAN PROGRAM

III.1. Analisis

III.1.1. Analisis Sistem Yang Sedang Berjalan Dan Yang Akan Dirancang

Pada sistem yang sedang berjalan saat ini, proses penyampaian pesan sangat rentan mengalami gangguan dari pihak lain yang tidak berkepentingan, hal ini disebabkan karena didalam proses penyampaian pesan khususnya pesan teks di dibarengi dengan penambahan sistem keamanan yang dapat menjaga kerahasiaan pesan yang akan disampaikan.

Sedangkan sistem yang dirancang, sebelum proses penyampaian pesan dilakukan, terlebih dahulu pesan yang ingin disampaikan akan melalui proses penyandian (enkripsi). Didalam proses enkripsi ini sendiri akan menggunakan metode *Vinegere Cipher*. Dimana metode *Vinegere Cipher* ini adalah metode penyandian teks dengan menggunakan deretan sandi Caesar berdasarkan huruf-huruf pada kata kunci.

III.1.2. Perangkat Lunak Yang Digunakan

Didalam proses perancangan aplikasi enkripsi dengan metode *Vinegere Cipher* ini akan menggunakan perangkat lunak sebagai berikut :

1. Menggunakan sistem operasi *Windows 7* dalam proses perancangan.
2. Menggunakan *Visual Studio 2008 (VB.Net 2008)*, didalam proses coding dan perancangan.

III.1.3. Sistem Operasi Yang Dipakai

Untuk menjalankan aplikasi hasil perancangan dibutuhkan spesifikasi berikut :

1. Komputer dengan spesifikasi minimum Intel Pentium IV 2,4 Ghz atau yang setara.
2. Ram minimum 1 Gb.
3. VGA Card standard.

III.1.4. Strategi Pemecahan Masalah

Didalam hal penyelesaian masalah yang akan diterapkan untuk meningkatkan sisi keamanan dalam proses pengiriman pesan adalah dengan melakukan penerapan kriptografi atau penyandian.

Metode yang akan diterapkan adalah metode *vigenere cipher*, dimana nantinya metode ini akan menggunakan sebuah kunci yang dipakai untuk menenkripsi dan mendekripsi pesan teks tersebut.

Didalam proses enkripsi pesan yang berupa *file text*, penggunaan aplikasi ini diwajibkan untuk memasukan kata kunci sebelum proses enkripsi dilakukan. Sedangkan dalam proses dekripsi pesan, pengguna juga wajib memasukan kata kunci yang sama dengan kata kunci pada saat melakukan proses enkripsi. Apabila kata kunci yang dimasukan tidak sesuai maka proses dekripsi tetap berjalan tetapi hasil dekripsi yang dihasilkan tidak sesuai dengan *plaintexts* semula. Hal ini dilakukan agar pihak lain yang tidak diperkenankan mendekripsikan pesan ini akan terkecoh.

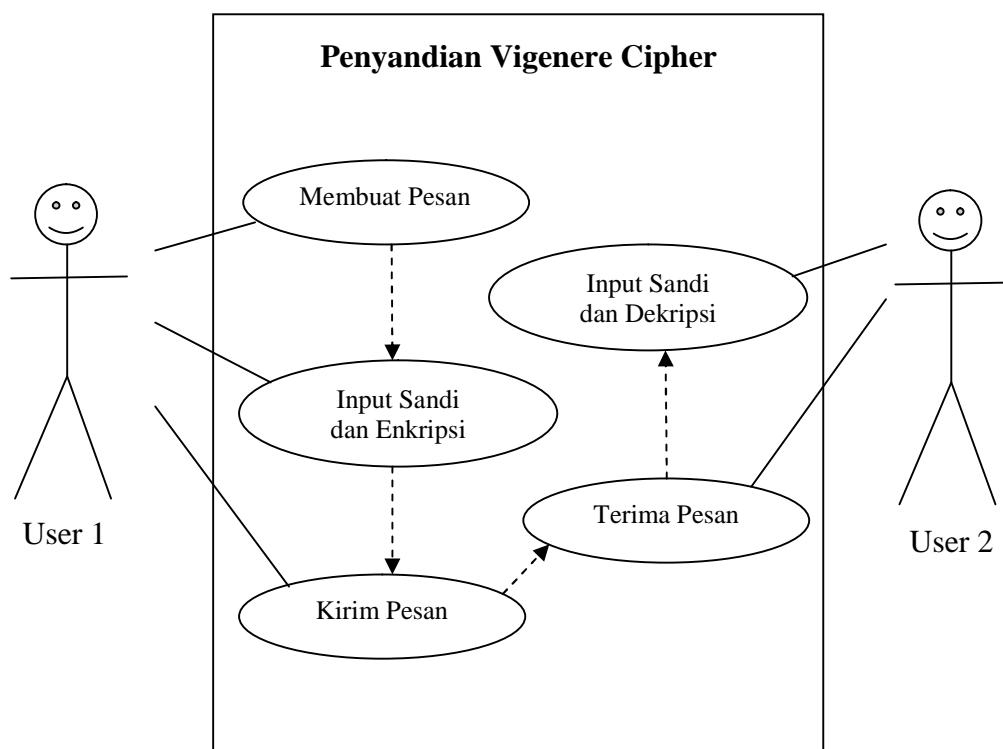
III. 2. Perancangan

III. 2. 1. UML Modeling

Sebelum melangkah kedalam tahap perancangan aplikasi lebih lanjut, maka dilakukan perancangan pemodelan visual dari aplikasi yang akan dibangun dengan menggunakan pemodelan UML (*Unified Modelling Language*), adapun pemodelan konsep perancangannya adalah sebagai berikut :

a. Usecase Diagram

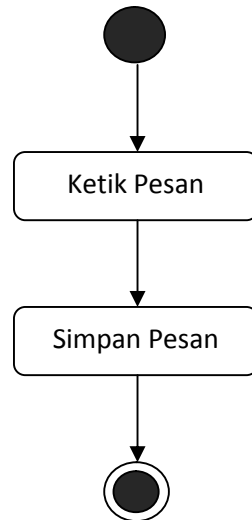
Berikut adalah usecase diagram dari aplikasi yang akan dirancang :



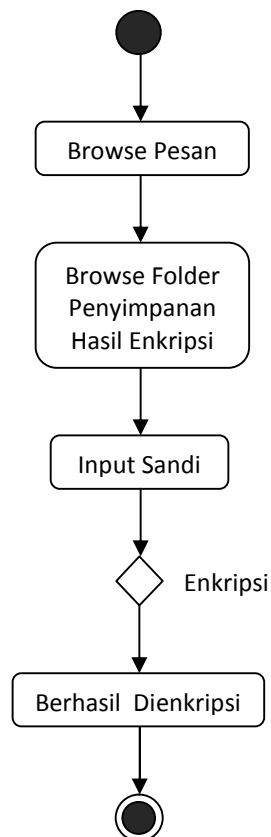
Gambar III.1. Usecase penyandian dengan *Vigenere Cipher*

b. Activity Diagram

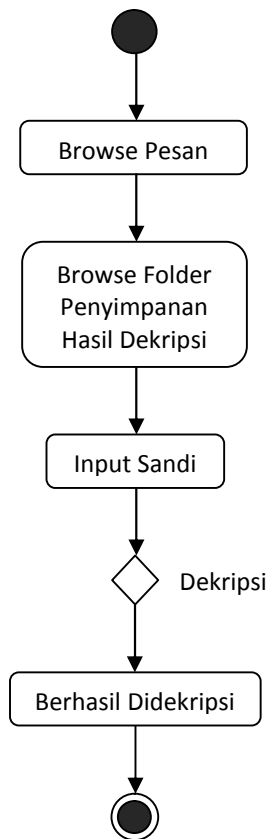
Berikut adalah activity diagram dari aplikasi yang akan dirancang :



Gambar III. 2. Activity Diagram Membuat Pesan



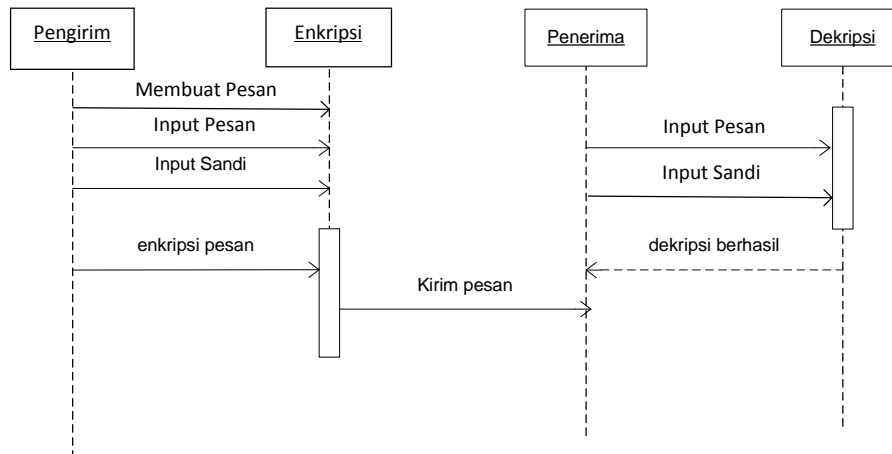
Gambar III. 3. Activity Diagram Enkripsi



Gambar III.4. Activity Diagram Dekripsi

c. Sequence Diagram Aplikasi Enkripsi dan Dekripsi

Berikut adalah sequence diagram dari aplikasi yang akan dirancang :



Gambar III. 5. Squence Diagram Aplikasi Enkripsi dan Dekripsi

III. 2. 2. Kelebihan dan Kekurangan Aplikasi

Dalam sebuah aplikasi tentunya memiliki kelebihan dan kekurangan, demikian pula halnya dengan aplikasi enkripsi dengan metode *Vigenere Cipher* ini. Berikut merupakan hasil dari kelebihan aplikasi yang telah dirancang :

1. Aplikasi mudah dipahami karena memiliki tampilan yang sederhana.
2. Tidak membutuhkan spesifikasi *hardware* yang tinggi.
3. Tidak membutuhkan proses instalasi untuk menggunakan aplikasi ini.
4. *File text* yang telah dienkripsi dapat langsung tersimpan pada *directory* yang diinginkan.
5. *File text* yang telah didekripsi kembali juga dapat langsung tersimpan pada *directory* yang diinginkan.

Sementara kekurangan yang dimiliki oleh aplikasi ini adalah :

1. Aplikasi hanya dapat melakukan proses enkripsi terhadap *file text* saja.
2. Hasil dekripsi juga hanya berbentuk *file text* saja.
3. Tidak dirancang untuk melakukan enkripsi terhadap jenis file lain.
4. Hanya dapat melakukan proses enkripsi dan dekripsi pada satu *file text* ingin dienkripsi atau didekripsi saja.
5. Belum terintegrasi secara jaringan.

III.2.3. Rancangan layar

Berikut adalah merupakan rancangan tampilan dari aplikasi yang akan dirancang :



Gambar III. 6. Tampilan dari halaman *Splash*

Tampilan diatas merupakan tampilan pembuka dari aplikasi, dimana dalam tampilan halaman pembuka terdapat *loading bar (progress bar)* yang menandakan aplikasi sedang melakukan proses *loading* aplikasi.

Gambar III.7. Tampilan halaman utama aplikasi dengan *tab menu* enkripsi

Tampilan diatas merupakan tampilan halaman utama dari aplikasi yang berfungsi untuk melakukan proses enkripsi, dimana dalam form aplikasi diatas tampak beberapa bagian yang memiliki fungsi sebagai berikut :

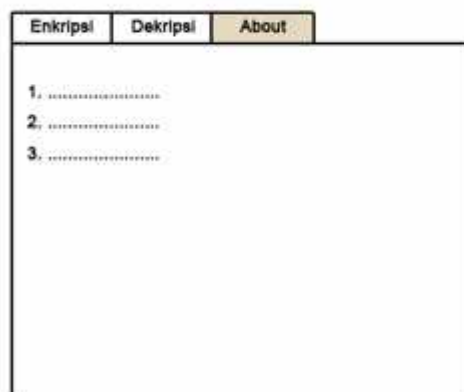
1. *Tab Menu* Enkripsi berfungsi sebagai halaman yang memberikan fasilitas proses enkripsi.
2. Tombol *Browse* pertama berfungsi untuk memanggil *file text* yang akan dienkripsi dan *directory* yang dipanggil akan terisi di *text box* yang ada di sampingnya.
3. Tombol *Browse* kedua berfungsi untuk mencari *directory* untuk menyimpan *file text* yang telah dienkripsi dan *directory*-nya akan terisi di *text box* yang ada di sampingnya.
4. *Text box* sandi berfungsi untuk menginputkan sandi yang diinginkan.
5. Tombol enkripsi berfungsi untuk memproses *file text* yang akan dienkripsi.

Gambar III.8. Tampilan halaman utama aplikasi dengan *tab menu* dekripsi

Tampilan diatas merupakan tampilan halaman utama dari aplikasi yang berfungsi untuk melakukan proses dekripsi, dimana dalam form aplikasi diatas tampak beberapa bagian yang memiliki fungsi sebagai berikut :

1. *Tab Menu* Dekripsi berfungsi sebagai halaman yang memberikan fasilitas proses dekripsi.
2. Tombol *Browse* pertama berfungsi untuk memanggil *file text* yang akan didekripsi dan *directory* yang dipanggil akan terisi di *text box* yang ada di sampingnya.
3. Tombol *Browse* kedua berfungsi untuk mencari *directory* untuk menyimpan *file text* yang telah dienkrpsi dan *directory*-nya akan terisi di *text box* yang ada di sampingnya.
4. *Text box* sandi berfungsi untuk menginputkan sandi yang sesuai dengan sandi pada saat pengenkripsian.

Tombol dekripsi berfungsi untuk memproses *file text* yang akan didekripsi.

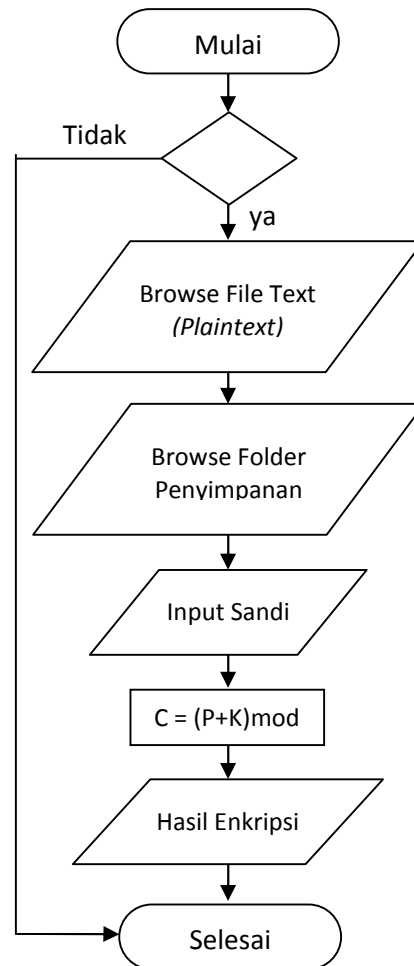


Gambar III.9. Tampilan halaman utama aplikasi dengan *tab menu about*

Tampilan diatas merupakan tampilan halaman utama dari aplikasi yang berisi informasi yang akan disampaikan oleh pembuat aplikasi tentan bagaimana cara menjalankan aplikasi.

III.2.4. Flowchart atau algoritma

Berikut adalah flowchart dari proses enkripsi dan dekripsi pada aplikasi yang akan dirancang :

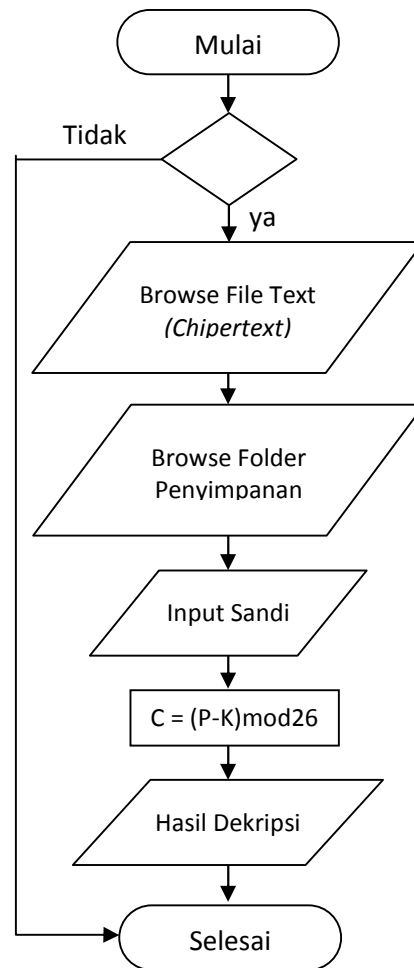


Gambar III.10. Flowchart proses enkripsi

Keterangan :

1. Mulai menyatakan awal dari rangkaian proses.
2. Kondisi melanjutkan proses atau keluar dari aplikasi

3. Dilanjutkan dengan menginputkan pesan yang berupa *file text* dengan mengklik tombol browse pertama.
4. Dilanjutkan dengan memilih *directory* yang akan menjadi tempat tersimpannya *file* yang telah dienkripsi dengan mengklik tombol browse kedua.
5. Dilanjutkan dengan mengetikkan kata sandi yang tersedia pada aplikasi
6. lalu untuk melakukan proses enkripsi maka user harus meng-klik pada tombol enkripsi.
7. Selanjutnya adalah aplikasi melakukan proses enkripsi.
8. Hasil akhir dari proses enkripsi berupa *file text* yang tersimpan pada *directory* yang telah *user* tentukan.
9. Selesai.



Gambar III. 11. Flowchart Proses Dekripsi

Keterangan :

1. Mulai menyatakan awal dari rangkaian proses.
2. Kondisi melanjutkan proses atau keluar dari aplikasi
3. Dilanjutkan dengan menginputkan pesan yang berupa *file text* yang telah dienkripsi *chipertext* dengan mengklik tombol browse pertama.

4. Dilanjutkan dengan memilih *directory* yang akan menjadi tempat tersimpannya *file* yang telah didekripsi dengan mengklik tombol browse kedua.
5. Dilanjutkan dengan mengetikkan kata sandi yang sesuai dengan kata sandi yang di-*input*-kan pada saat proses enkripsi.
6. Lalu untuk melakukan proses dekripsi maka user harus meng-klik pada tombol dekripsi.
7. Selanjutnya adalah aplikasi melakukan proses dekripsi.
8. Hasil akhir dari proses dekripsi berupa *file text* yang tersimpan pada *directory* yang telah *user* tentukan.
9. Selesai.