

BAB IV

PERANCANGAN PERANGKAT LUNAK

4.1 Metode Perancangan

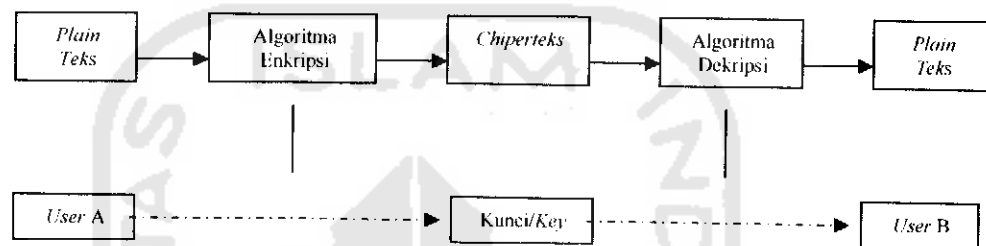
Tahap perancangan sistem dilakukan setelah tahapan analisis kebutuhan selesai dan didefinisikan secara jelas. Dalam tahap ini digambarkan lebih rinci berdasarkan tahapan sebelumnya, sehingga diperoleh algoritma dan detil aliran proses enkripsi dan dekripsi data yang akan dibuat. Adapun tujuan dan maksud dari tahap perancangan sistem adalah :

1. Untuk memenuhi kebutuhan kepada pemakai sistem
2. Untuk memberikan gambaran yang jelas dan rancang bangun yang diinginkan kepada pemrograman komputer.

Hasil dari tahapan perancangan ini harus dapat diterapkan menjadi prosedur dengan alat bantu bahasa pemrograman berorientasi object yaitu Borland Delphi 6. Metode perancangan yang digunakan yaitu memakai *flowchart* untuk mengetahui aliran setiap proses yang terjadi dalam proses enkripsi dan dekripsi *file* menggunakan algoritma RC6.

Block diagram dari program implementasi secara umum algoritma RC6 yang akan dibuat diperlihatkan pada gambar 4.1. Pertama-tama yang dilakukan adalah inialisasi data asli. Data ini akan disandikan/dienkripsi dengan menggunakan algoritma RC6 kemudian hasilnya akan di tampilkan, dari data yang telah disandikan akan dikembalikan ke data asli dengan proses dekripsi. Proses enkripsi/dekripsi memerlukan masukan/*input key* yaitu berupa karakter

ASCII. *Key input* digunakan untuk masukan baik sebelum enkripsi maupun dekripsi. Operasi *key* dimasukan sebagai kunci untuk proses dekripsi atau pengubahan data sandi ke data asli, data yang disandikan tidak dapat dikembalikan ke data asli jika kita tidak mempunyai kunci atau *key* yang pernah dimasukan pada waktu proses enkripsi.



Gambar 4.1 Rencana implementasi program secara umum

Program implementasi algoritma RC6 ini akan dibuat menjadi beberapa fungsi, antara lain:

- Enkripsi/dekripsi *file*
- Enkripsi/dekripsi teks

Implementasi program yang dibuat ditujukan untuk studi analisa algoritma RC6, jadi tampilan bersifat program pembelajaran RC6

4.2 Hasil Perancangan

Dari metode perancangan didapat hasil perancangan sistem yang meliputi program-program sebagai berikut:

1. Program enkripsi dan dekripsi *file*
2. Program enkripsi dan dekripsi teks

Rancangan program-program implementasi algoritma RC6 ini akan dibahas dibawah ini.

4.2.1 Proses Enkripsi / Deskripsi File

Pada proses ini berfungsi untuk enkripsi dan dekripsi *file*. Proses ini terdiri dari *browse file* yang akan dienkripsi, *save file* hasil enkripsi, *browse file* enkripsi yang akan dideskripsi, *save file* hasil dekripsi, tombol enkripsi *file*, tombol dekripsi *file*, *clear file*, chek boks untuk pilihan hapus *file*, serta informasi *file* yang terdiri nama *file* yang akan dienkripsi dan dekripsi, ukuran *file* dan waktu proses enkripsi dan dekripsi *file*. Subrutin program enkripsi dan dekripsi *file* ini:

1. Proses Enkripsi File

Proses ini merupakan proses untuk enkripsi *file*. Operasi program dimulai dari inialisasi masukkan program berupa *file* atau data elektronik kemudian pemberian *key* / kunci dan setelah proses enkripsi yang akan menyimpan *file* enkripsi tersebut kedalam *file* berekstensi *.cry

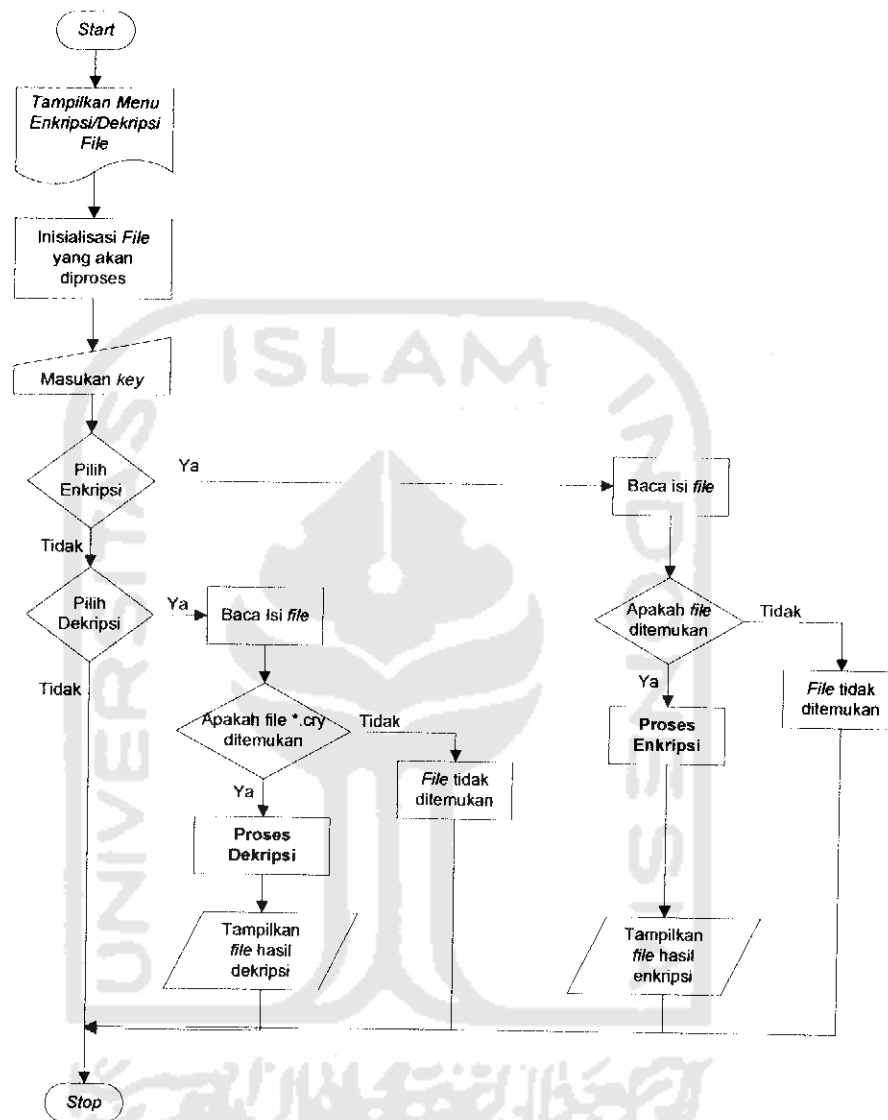
2. Proses Deskripsi File

Proses ini adalah proses untuk dekripsi yaitu membuka kembali *file* yang telah dienkripsi. Operasi program dimulai dari inialisasi *file* yang telah terenkripsi yaitu dengan format *.cry kemudian memasukkan *key* / kunci untuk membuka *file* enkripsi tersebut dan menyimpannya ke dalam *file* semula

3. Informasi waktu proses dan informasi ukuran *file*.

Waktu proses ditampilkan ketika mengenkripsi dan mendekripsi *file* yang memberikan informasi tentang kecepatan algoritma RC6, serta kapasitas *file*

yang dapat dilihat pada informasi *file* atau pada objek *triview*.



Gambar 4.2 Flowchart untuk enkripsi dan dekripsi file

4.2.2 Proses Enkripsi / Dekripsi Text

Proses ini berfungsi untuk enkripsi dan dekripsi *text*. Proses ini terdiri dari, tombol enkripsi, tombol dekripsi, tombol open *file*, tombol save *file*, tombol *clear*

text, tombol kembali ke *home* dan layar memo untuk mengisi karakter yang akan dienkripsi dan dekripsi. Dalam proses ini terdapat subrutin yaitu:

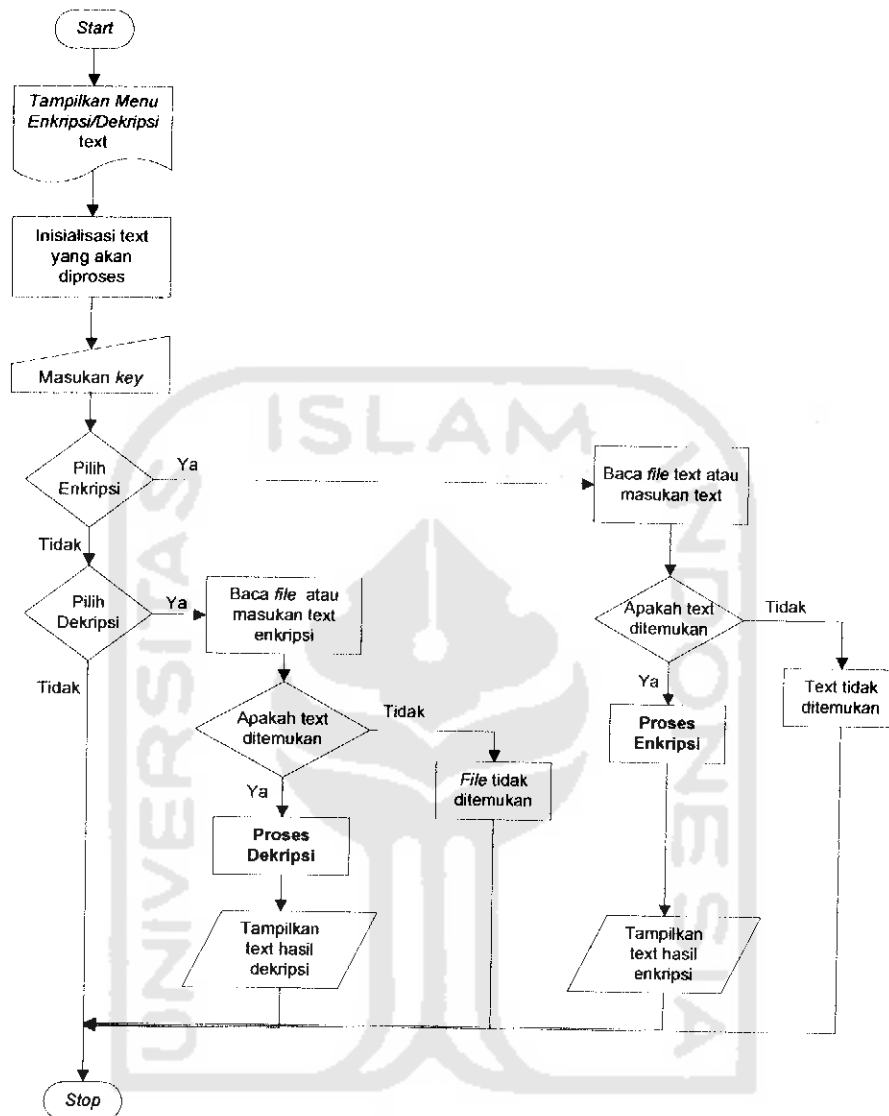
1. Proses Enkripsi *Text*

Proses ini merupakan program untuk enkripsi *text*. Operasi program dimulai dari inialisasi masukkan *text* pada memo *plaintext* kemudian pemberian *key*/kunci dan setelah proses enkripsi akan menampilkan karakter yang telah teracak dengan karakter lain baik berupa simbol atau karakter lain di memo *encrypttext* sehingga tidak dapat diartikan atau dipahami.

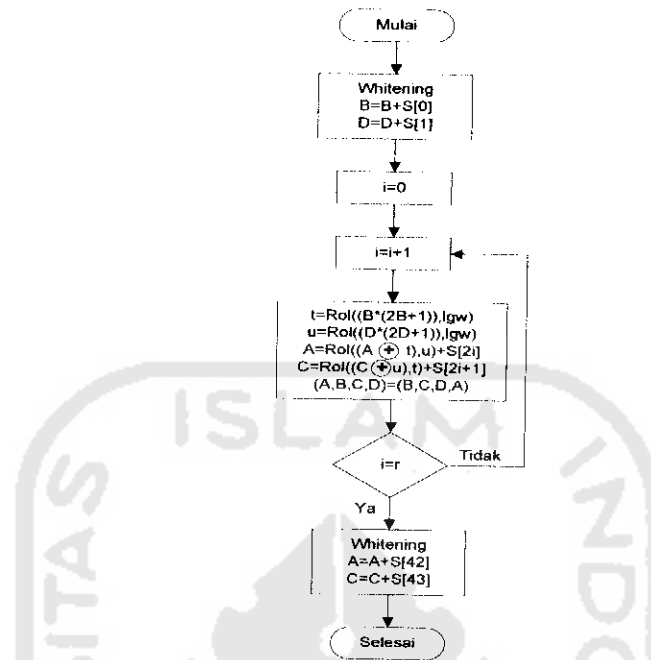
2. Proses Dekripsi *Text*

Proses ini adalah program untuk proses dekripsi yaitu mengembalikan data *text* yang tersandi menjadi data semula / asli sehingga dapat dipahami atau terbaca . Operasi program dimulai dari inialisasi *text* yang telah terenkripsi kemudian memasukkan *key*/kunci untuk proses dekripsi *text* dan menampilkannya ke memo *decrypttext*.

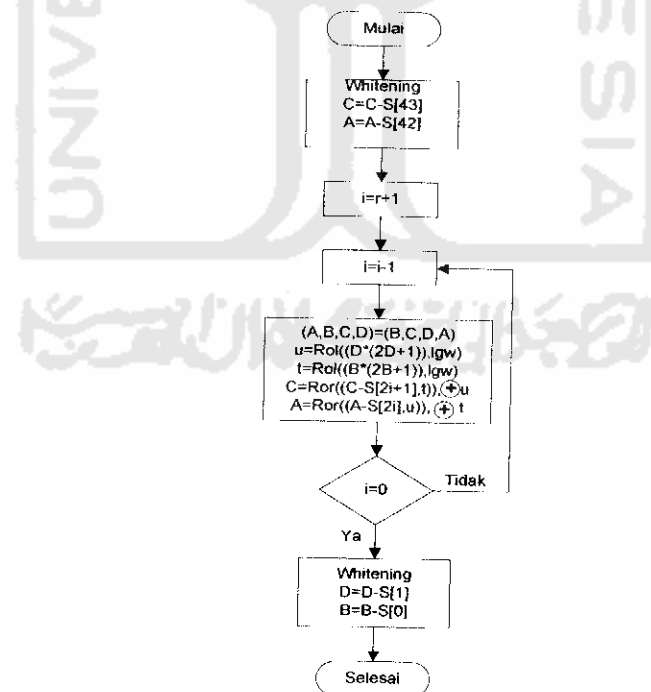
Flowchart dari program enkripsi dan dekripsi *text* dapat dilihat pada gambar 4.3.



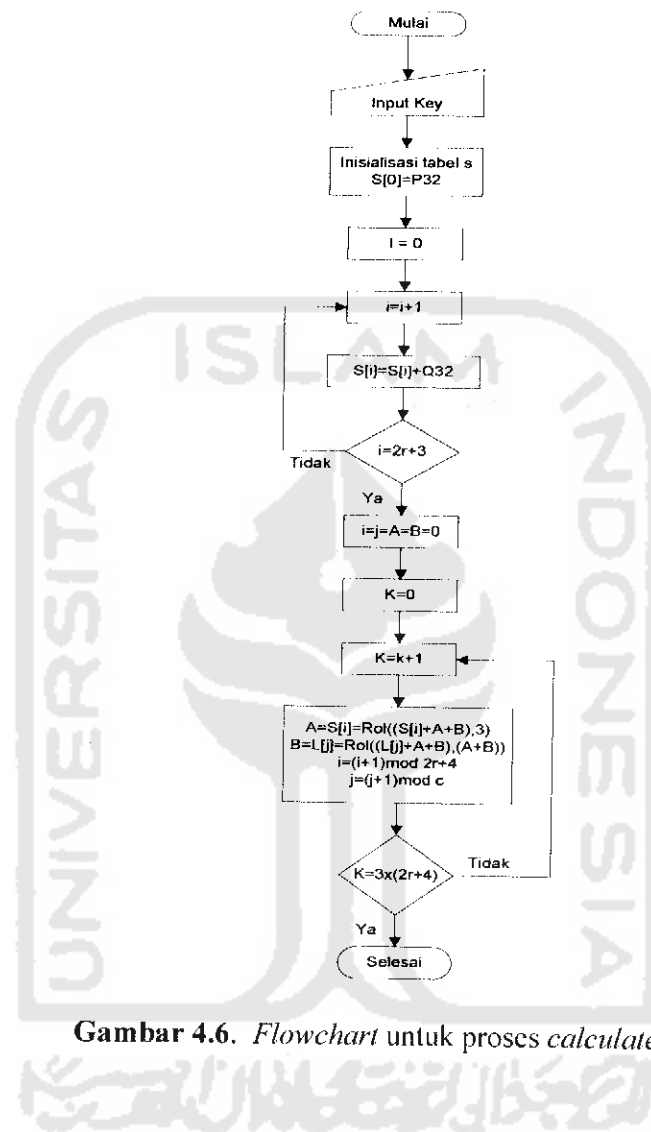
Gambar 4.3 Flowchart untuk enkripsi dan dekripsi text



Gambar 4.4 Flowchart untuk proses enkripsi file/text



Gambar 4.5. Flowchart untuk proses dekripsi file/text



Gambar 4.6. Flowchart untuk proses *calculate subkey*

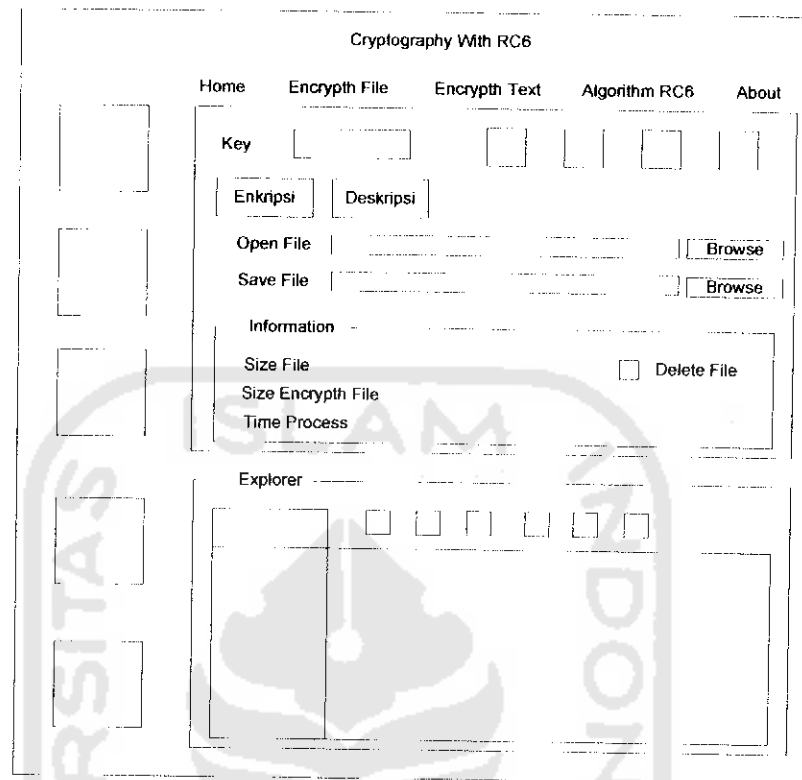
4.3 Perancangan Antar Muka (*Interface*)

Perancangan antarmuka dari sistem visualisasi ini menggunakan perancangan model grafis (*visual*). Antar muka dirancang sedemikian rupa sehingga diharapkan pengguna dapat berinteraksi secara baik dengan aplikasi. Antar muka yang baik haruslah dapat memberi petunjuk dan mengarahkan

pengguna dalam menggunakan aplikasi tersebut. Perancangan tersebut terdiri dari dua bagian, yaitu :

1. *Encrypt file* yaitu menu untuk mengenkrip dan mendeskrip *file*

Pada *interface* ini terdapat dua bagian, yaitu bagian untuk mengenkrip *file* dan bagian untuk mendeskrip *file*. Pada kedua bagian tersebut terdapat tombol-tombol yaitu : *browse file* yang akan dienkripsi, *save enkripsi file*, *browse file* yang akan didekripsi, *save dekripsi file*, tombol *encrypt file* yang berfungsi sebagai enkripsi *file*, tombol *decrypt file* yang berfungsi sebagai dekripsi *file*, tombol *clear file*, check boks untuk pilihan hapus *file*, tombol *exit* untuk keluar dari form *encrypt file*, *text key* untuk memasukkan kunci enkripsi/deskripsi dan informasi *file* yang terdiri dari kapasitas *file* original dan kapasitas *file* enkripsi serta waktu enkripsi dan dekripsi. Terdapat juga tampilan *explorer* untuk memudahkan dalam melihat letak *file* yang akan dienkrip/didekrip. Berikut gambar rancangan *interface* untuk enkripsi *file*.



Gambar 4.7 Antar muka untuk proses enkripsi/dekripsi *file*

2. Enkrip *text* yaitu menu untuk mengenkrip dan mendeskrip *text*

Pada *interface* ini terdapat beberapa tombol yaitu : tombol *encrypt text* yang berfungsi sebagai enkripsi *text*, tombol *decrypt text* yang berfungsi sebagai dekripsi *text*, tombol open tetx berfungsi untuk membuka *file* yang akan dienkrip/deskrip, tombol *save text* berfungsi untuk menyimpan *text* hasil enkrip/deskrip, tombol *clear* yang berfungsi untuk menghapus memo dan informasi *file*, tombol *exit* untuk keluar form, *text key* untuk *input* kunci enkripsi dan dekripsi, jumlah karakter yang akan dienkripsi dan yang telah terdekripsi serta waktu proses enkripsi dan dekripsi. Selain itu juga terdapat memo *plaintext* yaitu untuk *input text* yang akan dienkripsi, memo *encrypttext*

yaitu hasil dari proses enkripsi *text* dan memo *decryptext* yaitu hasil dari proses dekripsi *text*.

The image shows a web application interface for cryptography using the RC6 algorithm. The title is "Cryptography With RC6". The navigation menu includes "Home", "Encrypt File", "Encrypt Text", and "Algorithm RC6". The main content area contains several input fields: "Key" (a single character field), "Plaintext" (a large text area), "Ciphertext" (a large text area), "Information" (a text area), "Character Text" (a text area), and "Time Process" (a text area). A large watermark for "UNIVERSITAS ISLAM INDONESIA" is overlaid on the interface.

Gambar 4.8 Antar muka untuk proses enkripsi/dekripsi *text*