BAB III

ANALISA DAN DESAIN SISTEM

III.1. Analisa Masalah

Pembahasan yang akan diuraikan dalam sub bab ini meliputi gambaran hasil rancangan yang menjadi bagian-bagian komponen dengan tujuan mempelajari seberapa baik bagian-bagian komponen tersebut bekerja. Analisa masalah merupakan analisis pembahasan yang menjadi konsep perancangan yang akan dilakukan penulis. Kebutuhan manusia akan perangkat informasi dan komunikasi seakan menjadi kebutuhan yang tidak terpisahkan dalam kehidupan. Pemakaian *file* yang menjadi salah satu kebutuhan yang sering digunakan tetapi masih kurangnya pengamanan data *file* dengan enkripsi. Masih sedikitnya perancangan aplikasi menggunakan metode-metode pengamanan data gambar yang menggunakan algoritma DES dan *AES (Anvanced Encryption Standard).*

III.1.1. Analisa Perangkat Perancangan

Pada perancangan ini penerapan perangkat sebagai pendukung perancangan menggunakan beberapa perangkat yang dapat dijelaskan sebagai berikut :

1. Perangkat Lunak (*Software*), perangkat lunak merupakan perangkat yang digunakan untuk mendesain dan melakukan pemrograman, yang terdiri dari.
   a. *Operating SystemWindows Seven.*
   b. JDK *Java 1.7,* sebagai bahasa pemrograman *Java* dan *compiler Java*.
   c. *Netbeans 7.1.2,* sebagai *editor source code Java.*

2. Perangkat Keras (*Hardware*), perangkat keras merupakan perangkat yang digunakan untuk menjalankan dan implementasi aplikasi yang dirancang, yang terdiri dari.
   a. Komputer yang setara dengan *Intel pentium* 4.
   b. *Mouse, keyboard,* dan *Monitor.*

III.1.2. Analisa Metode Yang Digunakan

DES adalah blok cipher. Ini beroperasi pada blok dari 64-bit dalam ukuran. Sebuah masukan blok 64-bit dari plaintext akan dienkripsi menjadi 64-bit output blok teks cipher. Ini adalah sebuah Algoritma simetris, yang berarti algoritma dan kunci yang sama digunakan untuk enkripsi dan dekripsi. AES adalah salah satu algoritma populer yang digunakan dalam kriptografi kunci simetris. Algoritma AES menggunakan substitusi, permutasi, dan sejumlah putaran yang dikenakan pada tiap blok yang akan dienkripsi / dekripsi. Untuk setiap putarannya, menggunakan kunci yang berbeda. *Rijndael* beroperasi dalam orientasi *byte* sehingga memungkinkan untuk implementasi algoritma yang efisien ke dalam *software* dan *hardware*.

Keamanan DES terletak di kunci 56-bit. Blok plaintext diambil dan dimasukkan melalui permutasi awal. Kuncinya adalah juga diambil pada waktu yang sama. Kuncinya disajikan dalam blok 64-bit dengan setiap bit 8 menjadi cek paritas. Kunci 56-bit kemudian diekstraksi siap digunakan. 64-bit blok plaintext dibagi menjadi dua bagian 32 bit, bernama kanan setengah setengah kiri. Itu dua bagian dari plaintext kemudian digabungkan dengan data dari kunci dalam operasi yang disebut Fungsi F. Ada 16 putaran Fungsi f, setelah itu dua bagian yang

digabungkan menjadi satu 64-bit blok, yang kemudian dimasukkan melalui final permutasi untuk menyelesaikan operasi algoritma dan 64-bit teks cipher blok dikeluarkan.

Untuk algoritma AES, proses yang dilakukan untuk proses enkripsi pada program yaitu pengguna harus menginputkan file gambar yang akan di enkripsi. Kemudian pengguna memasukkan *key* untuk proses enkripsinya, lalu lakukan proses enkripsi file gambarnya. pada proses enkripsi akan dilakukan proses pertukaran *key* sebanyak jumlah *key* yang di masukkan, misalnyanya pada enkripsi AES dilakukan pertutukaran *key* sebanyak 16 karakter, kemudian hasil pertukaran *key* tersebut akan menghasilkan bentuk acak yang akan menambah jumlah karakter tersebut, pada konsep untuk enkripsi gambar, penulis akan merubah *pixel* gambar sesuai dengan menggunakan *key*, kemudian akan dilakukan proses pengacakan pixel pada gambar, hasil dari pengacakan pixel tersebut akan membuat gambar *file* asli tidak seperti aslinya lagi dan sulit untuk diketahui. Selanjutnya untuk proses dekripsi file gambar pengguna harus menginputkan gambar yang telah di enkripsi tersebut ke program aplikasi. kemudian masukkan *key* untuk dekripsi, lalu laukukan proses dekripsi gambar. Pada porses dekripsi program akan melakukan perhitungan algoritma yang akan membalikan fungsi *key* dari enkripsi, sehingga file gambar yang telah di enkripsi dapat berubah menjadi semula atau setidaknya file gambar dapat diketahui.

III.2. Desain Sistem

Desain sistem dibutuhkan sebagai gambaran langkah-langkah desain dan bagian-bagian yang dibutuhkan agar aplikasi dapat berjalan sesuai perancangan.

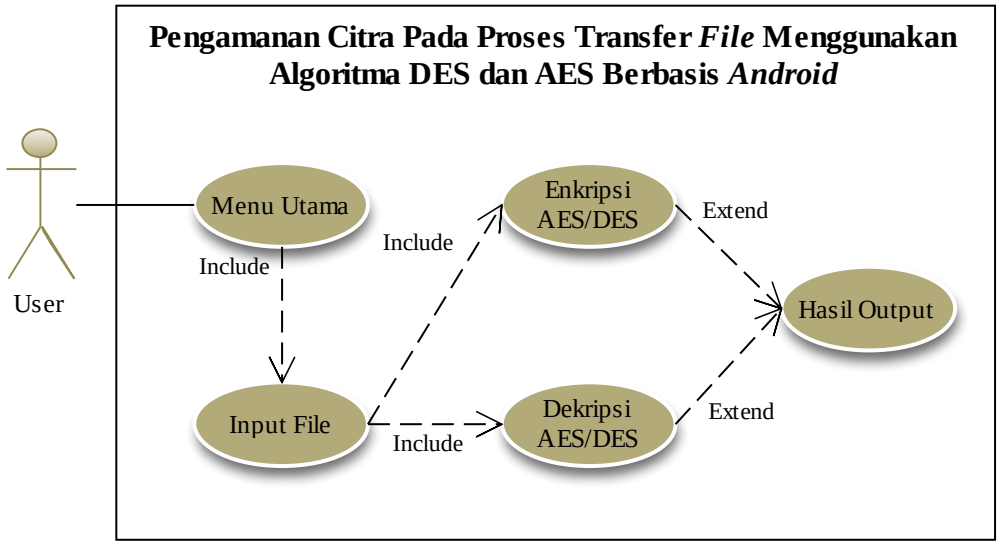
Pada tahapan ini menggambarkan digram alur kerja aplikasi dan desain *interface* yang akan dibuat. Adapun beberapa perancangan diagram dan desain yang akan dibuat dapat dijelaskan pada gambaran berikut ini.

III.2.1. *Flowchart* Rancangan Aplikasi

Agar dapat melihat struktur jalannya program maka dibuat *flowchart* (diagram alur). *Flowchart* digunakan sebagai dasar acuan dalam membuat program. Struktur program akan lebih mudah dibuat atau didesain. Selain itu juga jika terdapat kesalahan akan lebih mudah untuk mendeteksi letak kesalahannya serta untuk lebih memudahkan dalam menambahkan instruksi-instruksi baru pada program jika nantinya terjadi pengembangan pada struktur programnya.

III.2.2. *Use case* Diagram

*Use case*diagram menggambarkan aktor yang menggunakan aplikasi dan perilaku pengguna, seperti pada gambar III.1. di bawah ini.

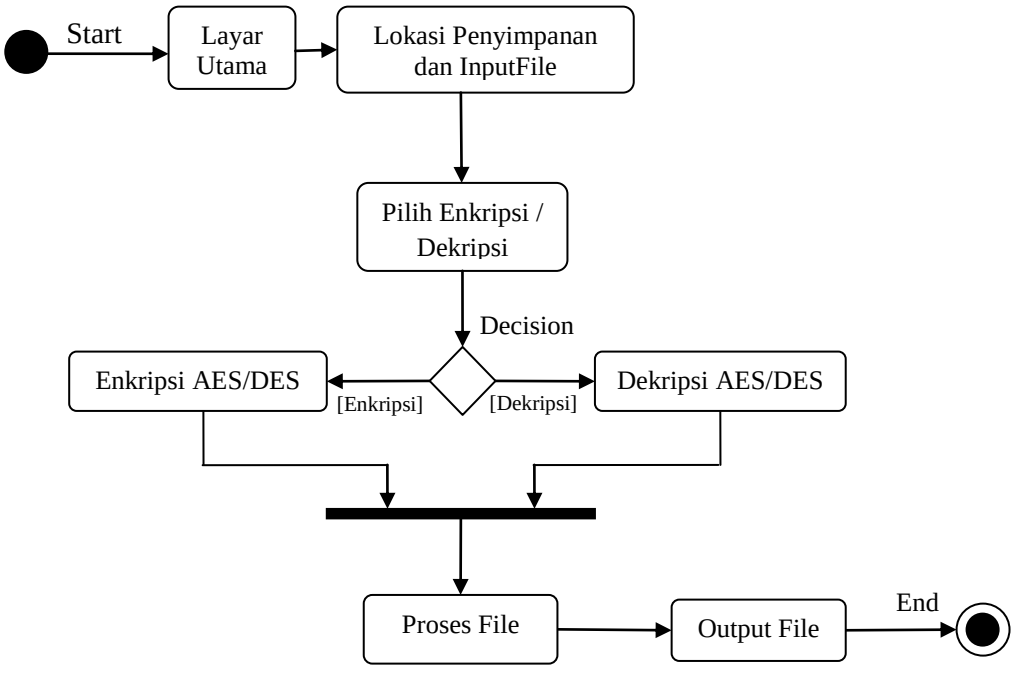


Gambar III.1. *Use Case* Diagram

Pada gambar III.1. di atas dapat dilihat proses yang berlangsung menunjukan tahapan penggunaan aplikasi yang dibangun. Pada tahapan ini penggunana dapat memulai dari menu utama aplikasi dan pengguna menginputkan file gambar lalu memilih proses enkripsi algoritma AES atau algoritma DES, begitu juga jika mendekripsi citra. Hasil dari proses merupakan *output file* yang telah diproses.

III.2.3.*Activity diagram*

Pada gambar dibawah ini adalah *activity*diagramaplikasi enkripsi dan dekripsi yang dirancang, dapat dilihat pada gambar III.2.



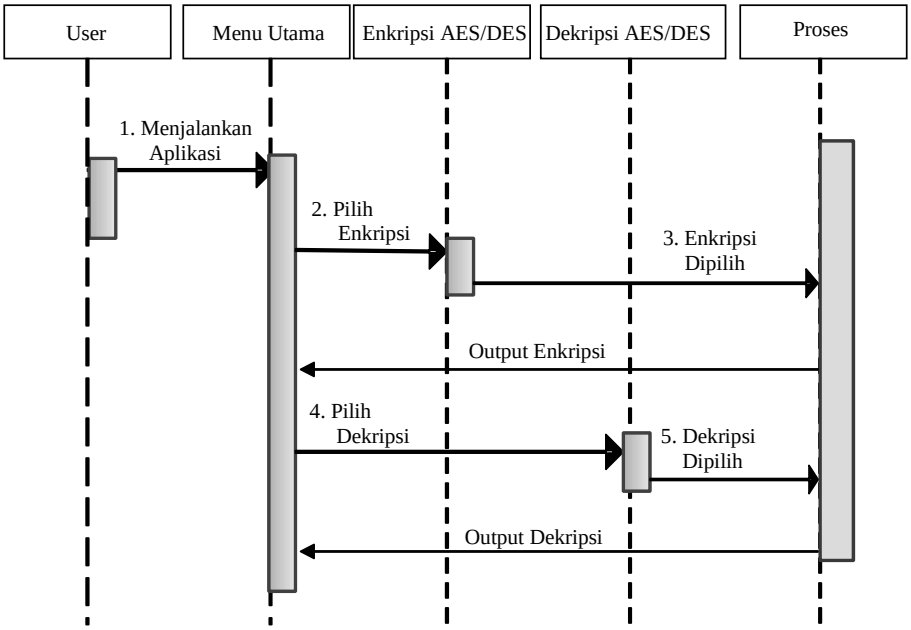
Gambar III.2. *Activity Diagram*

Pada gambar III.2. di atas menggambarkan *user* akan menemukan menu utama saat program dijalankan, *user* memilih proses yang ingin dilakukan yaitu enkripsi atau dekripsi AES atau DES lalu menentukan lokas *output file* dan *input*

file, setelah proses dijalankan secara otomatis aplikasi akan melakukan proses dan memberikan hasil atau *output* dilokasi yang telah ditentukan oleh *user*.

III.2.4.*Sequence* diagram

Sequence diagram menggambarkan kegiatan dari skenario penggunaan aplikasi. Adapun *sequence* diagram memilih proses pada aplikasi dapat dilihat pada gambar III.3. di bawah ini.

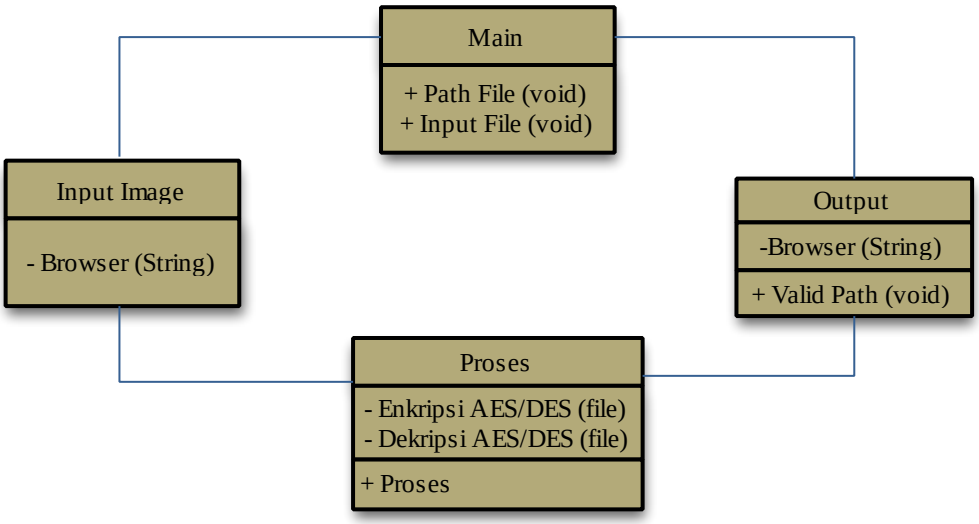


Gambar III.3. *Sequance* Diagram Pilihan Proses

Pada gambar III.3 di atas, Pengguna berinteraksi melalui pilihan proses yang ada pada menu utama, dapat dilihat pada *squence diagram* di atas, pengguna memilih proses yang disediakan yaitu proses enkripsi algoritma AES atau algoritma DES dan dekripsi algoritma AES atau DES. Setelah pilihan proses ditentukan oleh pengguna kembali pada menu utama.

III.2.5. *Class* Diagram

Untuk *class* diagram pada perancangan pengamanan citra pada proses transfer file ini, dapat dilihat pada gambar III.4. di bawah ini.



Gambar III.4. *Class* Diagram

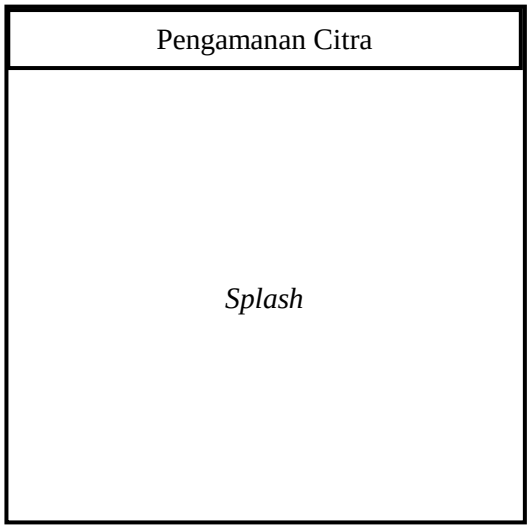
Class diagram adalah sebuah *class* yang menggambarkan struktur dan penjelasan *class*, paket, dan objek serta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi, dan lain-lain. *Class* diagram juga menjelaskan hubungan antar *class* dalam sebuah sistem yang sedang dibuat dan bagaimana caranya agar mereka saling berkolaborasi untuk mencapai sebuah tujuan.

III.3. Perancangan *Interface*

Perancangan *interface* adalah gambaran tampilan layar yang akan didesain, hal ini berguna agar proses perancangan dapat dilakukan sesuai desain yang telah dilakukan. Implementasi tampilan hasil program aplikasi yang telah dapat dijalankan harus sesuai dengan desain yang telah dibuat.

III.3.1. Rancangan *Interface Splash*

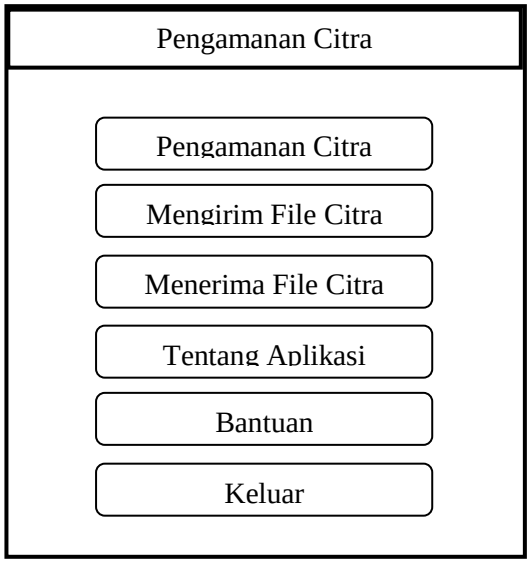
Rancangan splash ini merupakan tampilan yang terlihat sebelum masuk kedalam menu utama aplikasi. Adapun rancangan tersebut dapat dilihat pada gambar III.5 di bawah ini.



Gambar III.5. Rancangan *Interface Splash*

III.3.2. Rancangan *Interface* Menu *Utama*

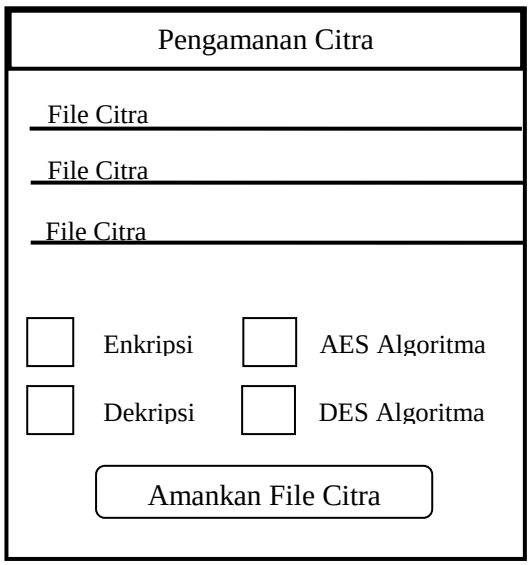
Rancangan Interface menu utama menjelaskan tampilan dimana terdapat menu dan fungsi *tools* yang digunakan untuk melakukan enkripsi maupun dekripsi citra yang dapat dilihat pada gambar III.6. dibawah ini.



Gambar III.6. Rancangan *Interface* Utama

III.3.3. Rancangan Interface Enkripsi/Dekripsi

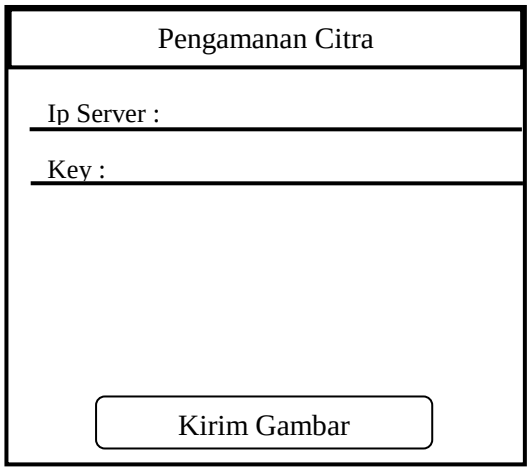
Interface enkripsi atau enkripsi ini merupkan tampilan yang dirancang dalam melakukan enkripsi citra ataupun dekripsi citra yang dapat dilihat pada gambar III.7. berikut.



Gambar III.7. Rancangan *Interface* Ekripsi/Dekripsi

III.3.4. Rancangan *Interface* Mengirim

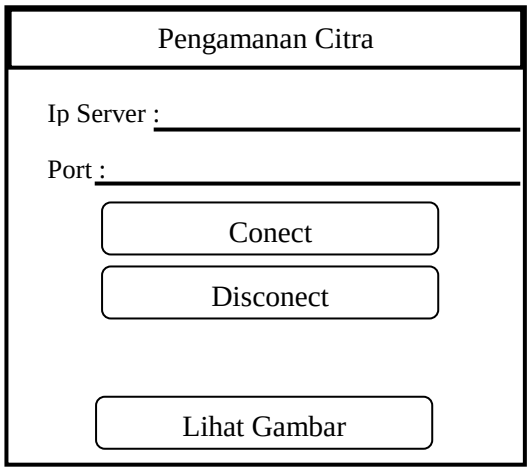
Pada desain *interface* ini berfungsi untuk mengirim *file* citra dengan menginputkan Ip server serta kunci *key* yang dapat dilihat pada gambar III.8.



Gambar III.8. Rancangan *Interface* Mengirim

III.3.5. Rancangan *Interface* Menerima

Pada desain *interface* ini berfungsi untuk mengirim *file* citra dengan menginputkan Ip server serta kunci *key* yang dapat dilihat pada gambar III.9.



Gambar III.9. Rancangan *Interface* Menerima

III.3.6. Rancangan *Interface* Bantuan

Pada desain tampilan *form* bantuan berfungsi untuk memberikan informasi bantuan penggunaan untuk pengguna. Dapat dilihat pada gambar III.10.

Gambar III.10. Rancangan *Interface* Bantuan

III.3.7. Rancangan *Interface* Tentang

Pada desain tampilan *form* tentang berfungsi untuk memberikan informasi perancangan aplikasi untuk pengguna. Dapat dilihat pada gambar III.11.

Gambar III.11. Rancangan *Interface* Tentang Aplikasi