KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN

SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER

PALCOMTECH

SKRIPSI

PEMANFAATAN ALGORITMA HUFFMAN DAN RC4 DALAM PENGAMANAN FILE (STUDI KASUS: FISIP UNIVERSITAS SRIWIJAYA PALEMBANG)

Diajukan Oleh:

1. **YUDI SEPTIANTO / 011170028**
2. **PUJIONO / 011170017**

Untuk Memenuhi Sebagian dari Syarat
Mencapai Gelar Sarjana Komputer

PALEMBANG

2021

KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN

SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER

PALCOMTECH

SKRIPSI

PEMANFAATAN ALGORITMA HUFFMAN DAN RC4 DALAM PENGAMANAN FILE (STUDI KASUS:FISIP UNIVERSITAS SRIWIJAYA PALEMBANG)

Diajukan Oleh:

1. **YUDI SEPTIANTO / 011170028**
2. **PUJIONO / 011170017**

Untuk Memenuhi Sebagian dari Syarat

Mencapai Gelar Sarjana Komputer

PALEMBANG

2021

KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN

SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER PALCOMTECH

HALAMAN PENGESAHAN PEMBIMBING SKRIPSI

NAMA	**: 1. PUJIONO / 011170017**
	2. YUDI SEPTIANTO / 011170028
PROGRAM STUDI	**: S1 INFORMATIKA**
JENJANG PENDIDIKAN	**: STRATA SATU (S1)**
JUDUL	**: PEMANFAATAN ALGORITMA HUFFMAN DAN RC4 DALAM PENGAMANAN FILE (STUDI KASUS : FISIP UNIVERSITAS SRIWIJAYA PALEMBANG)**

Tanggal: 5 Agustus 2021

Pembimbing

Guntoro Barovih, S.Kom., M.Kom.
NIDN: 0201048601

Mengetahui,

Ketua

Benedictus Effendi, S.T., M.T.
NIP: 09.PCT.13

KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
PALCOMTECH

HALAMAN PENGESAHAN PENGUJI SKRIPSI

NAMA	**: 1.PUJIONO**	**/ 011170017**
	2.YUDI SEPTIANTO	**/ 011170028**
PROGRAM STUDI	**: S1 INFORMATIKA**	
JENJANG PENDIDIKAN	**: STRATA SATU (S1)**	
JUDUL	**: PEMANFAATAN ALGORITMA HUFFMAN DAN RC4 DALAM PENGAMANAN FILE (STUDI KASUS : FISIP UNIVERSITAS SRIWIJAYA PALEMBANG)**	

Tanggal : 5 Agustus 2021	**Tanggal : 5 Agustus 2021**
Penguji 1	**Penguji 2**
Surahmat, S.T., M.T.	**Mahmud, S.Kom., M.Kom.**
NIDN : 0217058703	**NIDN : 0229128602**

Menyetujui,

Ketua

Benedictus Effendi, S.T., M.T.
NIP : 09.PCT.13

MOTTO DAN PERSEMBAHAN

MOTTO :

Berdoalah, Allah mendengarmu. Bersabarlah karena Allah akan menjawab doamu pada waktu yang tepat.

(pujiono)

Mempersembahkan kepada :

- Ayah dan Ibu yang selalu mendo'akan.
- Saudara-saudarku yang selalu Memberi semangat.
- Pembimbing yang selalu sabar saat membimbing.
- Teman-teman seperjuangan yang selalu mensupport.

KATA PENGANTAR

Puji dan syukur kehadirat Allah yang telah memberikan rahmat, kesehatan serta kesempatan sehingga penulis dapat menyelesaikan skripsi ini. Penelitian dengan judul **"Pemanfaatan Algoritma huffman dan RC4 dalam pengamanan file (Studi kasus : Fisip Univeristas Sriwijaya palembang)"** dan saya mengucapkan terima kasih kepada :

1. Ketua STMIK Palcomtech Palembang Bapak Benedictus Effendi, S.T., M.T.
2. Bapak Alfred Tenggono, S.Kom., M.Kom., selaku Ketua Program Studi S1 Informatika STMIK Palcomtech Palembang,
3. Bapak Guntoro Barovih, S.Kom., M.Kom., selaku pembimbing di STMIK Palcomtech Palembang,
4. Kedua orang tua, dan seluruh keluarga yang telah memberikan dukungan dan dorongan semangat dalam menyelesaikan laporan ini.

Akhir kata, semoga penulisan laporan skripsi ini bermanfaat bagi pihak yang membutuhkan.

Palembang, 26 juli 2021

Penulis

DAFTAR ISI

BAB III TINJAUAN PUSTAKA

BAB IV METODE PENELITIAN

DAFTAR GAMBAR

DAFTAR TABEL

DAFTAR LAMPIRAN

ABSTRACT

YUDI SEPTIANTO AND PUJIONO. *Utilization of Huffman and RC4 Algorithms in File Security (Case Study: FISIP, Sriwijaya University Palembang).*

The development of computer technology today produces an increasingly large output data size. On the other hand, data security and data confidentiality are one of the important factors that must be considered so that they are not known or manipulated by other parties. The movement of the combination of compression and encryption techniques is one solution that can be done to overcome this problem. Data compression is the process of converting an input data stream into a new data stream that has a smaller size. While RC4 encryption is a process that changes a code from being understandable to being unable to understand (unreadable). Decryption is a process with the same algorithm to restore the random information to its original form

Steganography is one of the encryption algorithms, where the encryption process inserts files into images, steganography is an algorithm that is set by the standard of modern encryption methods. In communication there are several forms ranging from using written or spoken. The increasing development of written communication makes the security and confidentiality aspects of data increasingly important. Cryptography is the science or art of maintaining data security by scrambling data or messages. While steganography is the science of hiding messages or data into a medium. The concepts of cryptography and steganography can be combined so as to produce a better message or data security system, with a steganographic algorithm technique using the and of file method. The results show that the encryption process is relatively faster.

Keywords: compression, encryption, huffman, RC4 and steganography

ABSTRAK

YUDI SEPTIANTO DAN PUJIONO. Pemanfaatan Algoritma Huffman dan RC4 Dalam Pengamanan File (Studi Kasus: FISIP Universitas Sriwijaya Palembang).

Perkembang teknologi komputer saat ini menghasilkan ukuran data output yang semakin besar, Disisi lain keamanan data dan kerahasiaan data merupakan salah satu faktor penting yang harus diperhatikan agar tidak diketahui atau dimanipulasi oleh pihak lain. Penggerakan penggabungan teknik kompresi dan enkripsi merupakan salah satu solusi yang dapat dilakukan untuk mengatasi masalah tersebut. Kompresi data adalah proses mengubah sebuah aliran data input menjadi aliran data baru yang memiliki ukuran lebih kecil. Sedanggkan enkripsi RC4 adalah suatu proses yang melakukan perubahan suatu kode dari yang bisa dimengerti menjadi tidak bisa mengerti (tidak terbaca). Dekripsi adalah suatu proses dengan algoritma yang sama untuk mengembaikan informasi teracak tadi menjadi bentuk aslinya

Steganografi adalah salah satu algoritma enkripsi, dimana proses enkripsi menyisispkan file kedalam gambar, steganografi merupakan algoritma yang ditetapkan standar metode enkripsi moderen. Dalam berkomunikasi ada beberapa bentuk mulai dari menggunakan tulisan maupun lisan. Semakin meningkatnya perkembangan komunikasi secara tulisan membuat semakin pentingnya aspek keamanan dan kerahasiaan data. Kriptografi merupakan ilmu atau seni untuk menjaga keamanan data dengan cara mengacak data atau pesan. Sedangkan steganografi adalah ilmu menyembunyikan pesan atau data ke dalam suatu media. Konsep kriptografi dan steganografi dapat dikombinakasikan sehingga menghasilkan sistem keamanan pesan atau data yang lebih baik, dengan Teknik algoritma steganografi menggunkan metode and of file Hasil penelitian menunjukkan bahwa proses enkripsi relatif lebih cepat.

Kata kunci : kompresi ,enkripsi, huffman, RC4 dan steganografi

BAB I

PENDAHULUAN

1.1 Latar Belakang

Seiring dengan perkembangan teknologi dan komunikasi yang begitu pesat, memudahkan kita untuk melakukan pertukaran dengan data orang lain secara tepat. Namun terkadang keamanan dalam pertukaran data tersebut kurang disadari oleh kita sehingga terjadi pencurian data. Keamanan data merupakan hal penting harus ada dan diterapkan untuk menjaga data tetap aman dan tidak disalahgunakan oleh pihak yang tidak bertanggung jawab. Dengan adanya pencurian data maka aspek keamanan dalam pertukaran informasi serta penyimpanan data dianggap penting.

Fakultas Ilmu Sosial Dan Ilmu Politik (FISIP) Universitas Sriwijaya saat ini telah mejadi aset penting yang mana di Universitas ini sendiri memiliki informasi dan *file-file* yang sangat rahasia. Fakultas Ilmu Sosial Dan Ilmu Politik (FISIP) Universitas Sriwijaya merupakan kampus yang memiliki jurusan Adminitrasi *public*, Sosiologi, Ilmu Komunikasi, Hubungan internasional. khususnya pada admin setiap jurusan menyimpan data berkas *file* soal yang harus terjaga keamananya dari pihak yang tidak bertanggung jawab. banyak *file* soal yang bersifat rahasia dan tidak bisa dipergunakan atau dirubah oleh pihak yang tidak berhak. Untuk mengamankan *file* maka dapat menggunakan Pemanfaatan algoritma huffman dan RC4 dalam pengamanan file. Oleh karena itu, pengguna *file* soal membutuhkan bantuan

untuk keamanan akan *file* soal yang disimpan. Penerapan Algoritma Huffman dan RC4 dalam pengamanan *file* pada FISIP Universitas Sriwijaya akan difokuskan bagaimana dapat mengamankan *file* soal yang tersimpan menjadi aman sampai dengan dokumen dibuka oleh pihak yang berhak untuk membukanya.

Berdasarkan uraian diatas, penelitian ini adalah untuk mengetahui bagaimana implementasi dari gabungan antara kedua algoritma tersebut, selain itu, penelitian ini juga bertujuan untuk mengetahui bagaimana rasio perbandingan ukuran *file* antara *file* awal dan *file* yang terkompresi. Tahapan kompresi digunakan untuk proses pemampatan, dan tahapan dekompresi untuk proses pengembalian *file* ke bentuk dan ukuran yang semula.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dikemukakan, maka perumusah masalah yang dirumuskan adalah Bagaimana mengimplementasika perangkat lunak yang dapat mengenkripsi dan mendekripsi data dengan menggunakan algoritma Huffman dan RC4 pada Fakultas Ilmu Sosial Dan Ilmu Politik Universitas Sriwijaya.

1.3 Batasan Masalah

Beberapa batasan masalah dalam penelitian ini adalah sebagai berikut:

1. Dapat memberikan layanan proses enkripsi (pengacakan data)
2. Dapat memberikan layanan proses penyisipan pesan rahasia kedalam gambar

3. Untuk mempercepat enkripsi dan dekripsi file hanya dibatasi sebesar 2MB.
4. Dapat memberikan tujuan layanan proses dekripsi (pengembalian data seperti semula).
5. Format yang bisa di enkripsi dan dekripsi berupa doc, xls, txt,pdf,ppt dan jpg/jpeg
6. Aplikasi yang di buat berbasis web dengan menggunankan bahasa pemograman PHP dan MySQL sebagai penyimpanan.

1.4 Tujuan dan Manfaat Penelitian

1.4.2. Tujuan Penelitian

Adapun tujuan dilakukan pembuatan laporan penelitian proposal skripsi ini adalah agar dapat membantu pihak Admintrasi Fakultas Ilmu Sosial dan ilmu Politik (FISIP) Universitas Sriwijaya Dalam menyimpan data informasi *file-file* rahasia.

1. Memudahkan untuk menjaga keamanan data melalui proses enkripsi.
2. Menghindari pencurian data dengan menggunakan kriptografi supaya data tidak dapat disalah gunakan dengan orang yang tidak berhak.
3. Memaksimalkan proses enkripsi dan dekripsi secara optimal.

1.4.2. Manfaat Penelitian

1.4.2.1. Manfaat Bagi Penulis

Manfaat yang di perolah mahasiswa dari penelitian ini adalah

1. Dapat mengaplikasikan ilmu pengetahuan yang di dapat

selama perkuliahan pada bidang pemograman dan dapat menambah ilmu.

2. Melatih dan Menabah pengalaman serta meningkatkan keterampilan penulis dalam melakukan pekerjaan sebagai bekal dalam memasukin dunia kerja.
3. Memberikan manfaat berupa bertambahnya ilmu pengetahuan, pengalaman, pengenalan dan pemahaman berdasarkan penelitian yang telah dilakukan.
4. Memberika informasi dan masukan serta dapat memberikan pengetahuan dan motivasi kerja terhadapat kinerja karyawan di instansi tersebut.
5. Dapat menambah ilmu pengetahuan pengguna dalam keamanan data dan dapatmembantu pengguna untuk melakukan

1.4.2.2 Manfaat Bagi Akademik

Manfaat yang diperoleh akademik dari penelitian ini adalah:

1. Sebagai bahan referensi bagi penulis lain untuk dijadikan perbandingan dalammenyusun proposal dan skripsi pada penelitian selanjutnya.
2. Sebagai bahan evaluasi sejauh mana kemampuan mahasiswa dalam menerapkanilmu pengetahuan yang telah diberikan.
3. Dapat digunakan sebagai tambahan ilmu bagi mahasiswa yang melakukan kajian terhadap algoritma Huffman dan RC4 di masa yang akan datang.

4. Dapat memberikan informasi ilmu untuk penelitian selanjutnya yang berkaitandalam bidang keamanan data.

1.4.2.3 Manfaat Bagi Universitas

Manfaat yang diperoleh universitas dari penelitian ini adalah universitas mendapatkan sebuah sistem yang bisa memperkecil data dari data aslinya, Untuk mempercepat enkripsi dan dekripsi file hanya dibatasi sebesar 2MB dan Format file yang bisa dienkripsi dan dideskripsi hanya, pdf, docx, xls, txt,ppt, dan jpg/jpeg.

1.5. Sistematika Penulisan

Sistematika penulisan dalam skripsi ini di buat untuk dapat memberikan gambaran secara garis besar mengenai isi dari penulisan skripsi ini. Sistematika penulisan dalam skripsi ini adalah sebagai berikut:

BAB I PENDAHULUAN

Bab ini berisi pemaparan mengenai latar belakang, rumusan masalah, batasan masalah, tujuan penelitan, manfaat penelitian dan sistematika penelitian.

BAB II GAMBARAN UMUM PERANGKAT *LUNAK*

Bab ini berisi pemaparan mengenai perangkat *lunak* yang akan dikembangkan dalam penelitian.

BAB III TINJAUAN PUSTAKA

Bab ini berisi pemaparan mengenai teori-teori yang melandasi dilakukannya penulisan yang terdiri dari landasan teori, penelitian terdahulu dan kerangka penelitian.

BAB IV METODE PENELITIAN

Bab ini berisi pemaparan mengenai waktu dan jenis penelitian, jenis dan pengumpulan data, pengembangan dan pengujian sistem.

BAB V HASIL DAN PEMBAHASAN

Bab ini merupakan bab terakhir atau bab penutup. Bab ini berisi pemamparan mengenai suatu hasil yang diperoleh dalam penelitian dan pembahasan serta ditemukan peneliti.

BAB VI PENUTUP

Penjelasan hasil dari penelititan yang terdiri dari kesimpulan dan saran.

BAB II

GAMBARAN UMUM PERANGKAT LUNAK

2.1. Kompresi Data

Menurut Nurhardian dan Pudoli, (2016) kompresi data merupakan suatu teknik untuk memperkecil jumlah ukuran data (hasil kompresi) dari data aslinya. Pemampatan data umumnya diterapkan pada mesin komputer, hal ini dilakukan karena setiap simbol yang dimunculkan pada komputer memiliki nilai bit-bit yang berbeda. Misal pada ASCII setiap simbol yang

dimunculkan memiliki panjang bit 8 bit, misal kode A pada ASCII mempunyai nilai desimal = 65, jika dirubah dalam bilangan biner menjadi 010000001. Pemampatan data digunakan untuk mengurangkan jumlah bit-bit yang dihasilkan dari setiap simbol yang muncul. Dengan pemampatan ini diharapkan dapat mengurangi (memperkecil ukuran data) dalam ruang penyimpanan

2.2. Fenomena

Penerapan metode pengamanan data enkripsi dan dekripsi ini penulis menggunakan Algoritma Rivest Code 4 (RC4) merupakan salah satu algoritma kunci simetris dibuat oleh RSA Data Security Inc (RSADSI) yang berbentuk *stream chipper*. Algoritma ini ditemukan pada tahun 1978 oleh Ronald Rivest dan menjadi symbol keamanan RSA (merupakan singkatan dari tiga nama penemu: Rivest Shamir Adleman). RC4 menggunakan Panjang kunci dari 1 sampai 256 *byte* yang digunakan untuk menginisialisasikan tabel sepanjang 256 *byte.* Tabel

ini digunakan untuk generasi yang berikut dari *pseudo random* yang menggunakan XOR dengan plainteks untuk menghasilkan cipherteks. Masing-masing elemen dalam tabel saling ditukarkan minimal sekali.

RC4 merupakan salah satu jenis *stream cipher* sehingga RC4 memproses unit atau input data, pesan atau informasi pada satu saat. Unit atau data pada umumnya sebuah *byte* sehingga dengan car aini enkripsi atau dekripsi dapat dilaksanakan pada Panjang variable. Algoritma ini tidak harus menunggu sejumlah input data, pesan atau informasi tertentu sebelum diproses, atau menambahkan *byte* tembahan untuk mengenkrip.

Algoritma RC4 menggunakan dua buah S-box yaitu *array* sepanjang 256 yang berisi permutasi dari bilangan 0 sampai 255, dan S-box kedua, yang berisi permutasi merupakan fungsi dari kunci dengan Panjang yang variable. Secara umum, algoritma RC4 terbagi menjadi dua, inisialisasi *state-array* dan penghasilan kunci enkripsi serta pengenkripsiannya.

BAB III

TINJAUAN PUSTAKA

3.1. Deskripsi Mengenai RC4

Menurut Nurhadian dan Pudoli, (2016) Nurhardian dan Pudoli, (2016) RC4 merupakan metode penyandian pesan teks yang melakukan enkripsi per bit sehingga kelebihan dari metode ini kerusakan pada satu bit tidak mempengaruhi keseluruhan isi pesan. Pada RC4 dihasilkan pseudo random stream bit. Seperti halnya stream cipher lainnya, algoritma RC4 ini dapat di gunakan untuk mengenkripsi dengan mengkombinasikannya dengan plainteks menggunakan Excusive-or (Xor). Untuk proses dekripsidilakukan cara yang sama dengan kunci yang sama, karena Xor merupakan fungsi simetrik.

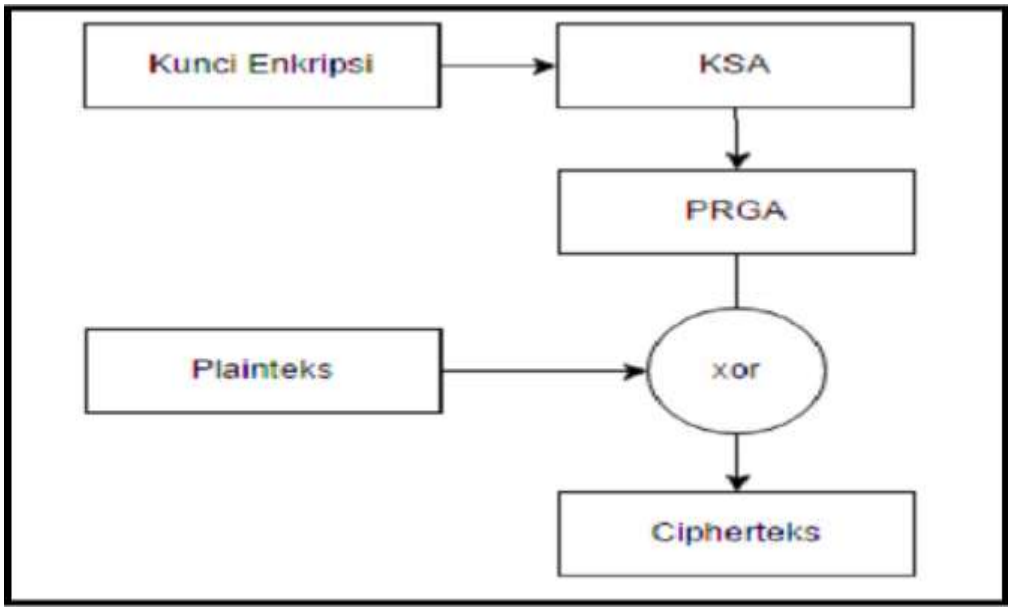
Secara garis besar proses algoritma RC4 dibagi menjadi dua bagian, yaitu Key Scheduling Algorithm (KSA) dan Pseudo Random Generation Algorithm (PRGA)

3.1.1 Algoritma Enkripsi RC 4

Menurut Nurhadian dan Pudoli, (2016) RC4 adalah *cipher* aliran yang digunakan secara luas pada sistem keamanan seperti protokol SSL (*Secure Socket Layer*). Algoritma kriptografi ini sederhana dan mudah diimplementasikan. RC4 dibuat oleh Ron Rivers dari Laboratiorium RSA (RC adalah singkatan dari Ron's *Code*). RC4 membangkitkan aliran

kunci (*keystream*) yang kemudian di-XOR-kan dengan *plaintext* pada

waktu enkripsi (atau di- XOR-kan dengan *bit-bit ciphertext* pada waktu dekripsi) untuk menujukan proses enkripsi dari algoritma RC4, berikut dapat dilihat pada Gambar 3.1 Di bawah:



Gambar 3.1 Arsitektur Enkripsi RC4

1. Inisialisasi array S-box pertama, S [0], S[1],.....,S[255]. Diisi dengan bilangan 0 sampai 255 sehingga array S-box aray S bentuk S[0]=0, S[1]=1, , S [255]=255

   Forr = 0 to 255

   S[r] = r

2. Inisialisai array kunci (S- box lain), misal array kunci k dengan panjang 256 jika panjang kunci K <256, maka dilakukan padding yaitu penambahan byte sehingga panjang kunci menjadi 256 byte misalnnya K = “abc” yang hanya terdiri dari 3 byte (3 huruf), maka lakukan padding dengan penambahan byte (huruf) semu, misalnya

K

= “abcabcabcabcabc..” sampai panjang K mencapai 256 byte sehingga S-box array kunci K berbentuk K [0], [1], ,K[255]

For i = 0 to 255

K[i] = kunci [i mod lenght];

3. Permutasi terhadap nilai-nilai didalam array S dengan cara menukarkan isi array S[I] dengan S[j], prosesnya adalah sebagai berikut

J = 0
For i = 0 to 255
J = (j+S[i] + k(j) mod 256
Isi S[i] dan isi S[j] ditukarkan

4. Membangkitkan aliran kunci (keystream) selanjutnya digunakan untuk enkripsi.

i = j = 0

i = (i + 1) mod 256

j = (j +S[j] mod 256

isi S[i] dan isi S[j] ditukar

t = (S[i] + S[j] mod 256

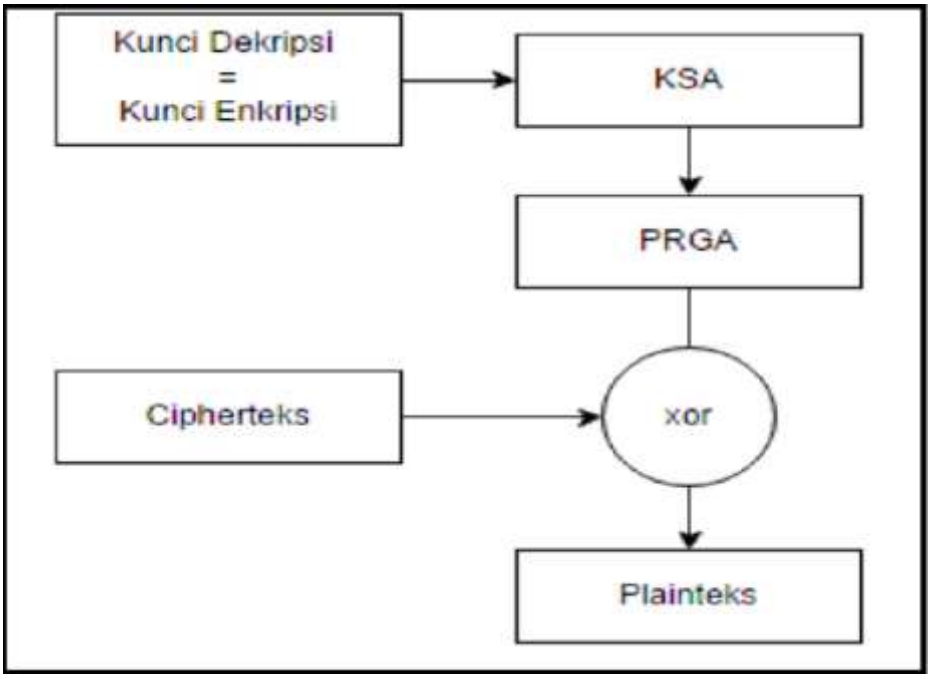
K = S[t];

5. Kunci aliran k kemudian digunakan untuk mengenkripsi plaintext ke-idx sehingga didapatkan ciphertext, sedangkan untuk mendapatkan palintext dengan cara chipertext di- XOR- kan

dengan kunci yang sama dengan proses enkripsi.

3.1.2 Algoritma Dekripsi RC4

Menurut Nurhardian & Pudoli, (2016) Algoritma dekripsi RC4 sama dengan algoritma enkripsinya, perbedaannya hanya pada saat stream generation, yaitu untuk menghasilkan plainteks semula, maka ciphertext nya akan dikenakan operasi XOR terhadap pseudorandom bytenya. Algoritma key setup pada proses dekripsi sama dengan algoritma enkripsinya yang diproses inisialisasi S-Box, penyimpanan kunci kedalam key bytearray hingga proses inisialisasi S-Box berdasarkan key byte array nya. Untuk itu proses dekripsi dan enkripsi akan menghasilkan key stream yang sama. Perbedaannya hanya pada stream generationnya, yaitu yang dioperasikan bersama key stream adalah ciphertext untuk menghasilkan kembali plaintext. Berikut ini akan diberikan Gambar proses dari dekripsi RC4. Lihat Gambar 3.2



Gambar 3.2 Arsitektur Dekripsi RC4

3.2 Algorimta Huffman

Menurut Nurhadian dan Pudoli, (2016) Pengkodean dengan metode Huffman dibangun dari panjang variabel kode-kode yang disusun dari bit-bit. Simbol dengan probabilitas yang tinggi akan memperoleh kode-kode paling pendek sedangkan simbol dengan probabilitas paling rendah akan memperoleh kode terpanjang. Contoh untuk string 'NURHARDIAN' mempunyai panjang bit sebanyak 80 bit karena 1 karakter dikodekan dengan 8 bit (ASCII) akan diperoleh jumlah bit untuk tiap simbolnya dengan jumlah yang lebih sedikit atau bitnya lebih pendek yaitu 27 bit, sehingga secara otomatis ukuran filenya berkurang.

Kode Huffman digunakan secara luas dan sangat efektif untuk kompresi data. Menurut (Nurhadian dan Pudoli, 2016) bisa menghemat 20%-

90% dari ukuran semula, tergantung tipe karakter yang dikompresi. Algoritma huffman menggunakan tabel yang menyimpan frekuensi kemunculan dari masing-masing simbol yang digunakan dalam file tersebut dan kemudian mengkodekannya dalam bentuk biner

3.2.1 Algoritma kompresi Huffman

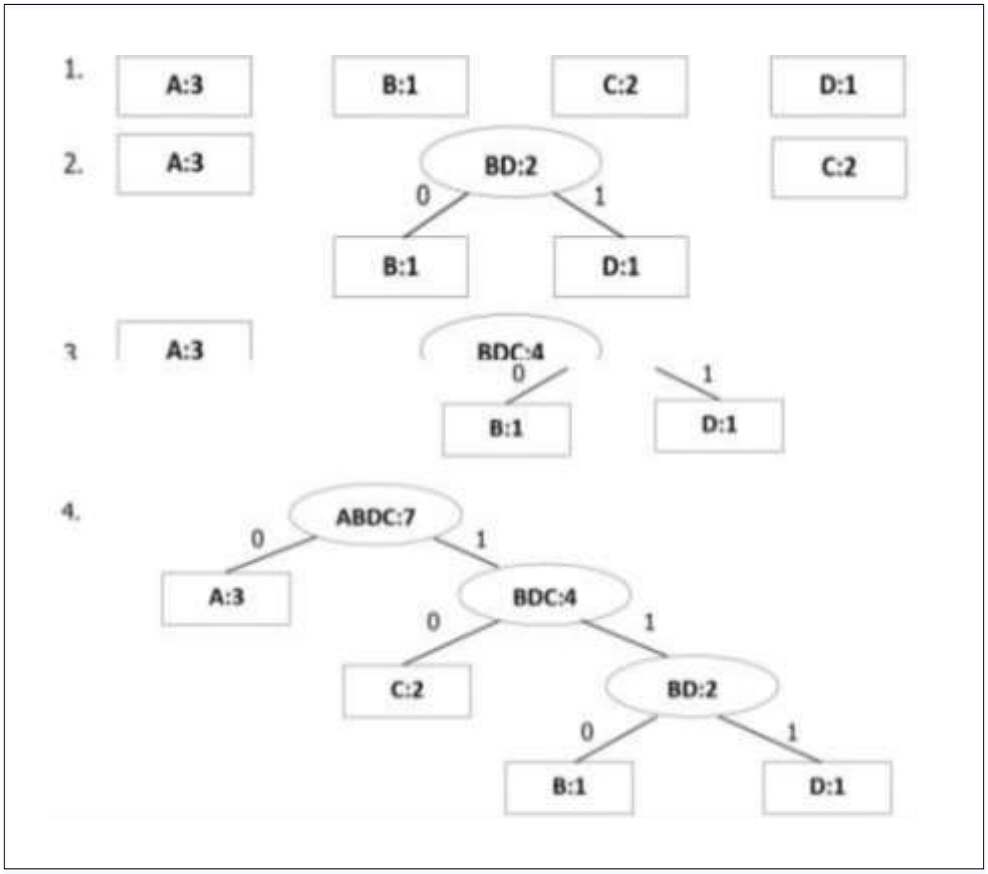
Algortima Huffman ditentukan oleh seorang mahasiswa MIT pada tahun 1952 bernama David Huffman. Algoritma ini termasuk dalam metode kompresstatistic yang memanfaatkan perhitungan statiska (*Statiscal Methods*) untuk melihat probabilitas kemunculan data dari sebuah dokumen. Probabilitas tersebut digunakan untuk menentukan cara untuk mengolah data terebut agar bisa dipadatkan.

3.2.2 Pembentukan Pohon Huffman

Menurut Kusniyati *et.al*, (2018) Kode Huffman pada dasarnya merupakan kode prefik (prefix code). Kode prefix adalah himpunan yang berisi sekumpulan kode biner, dimana pada kode prefix ini tidak ada kode biner yang menjadi awal bagi kode biner yang lain. Kode prefix biasanya dipresentasikan sebagai pohon biner yang diberikan nilai atau label. Untuk

cabang kiri pada pohon biner diberi label 0, sedangkan cabang kanan pada pohon biner diberi label 1. Rangkaian bit yang terbentuk pada setiap lintasan dari akar ke daun merupakan kode prefix untuk karakter yang berpadanan. Pohon biner ini biasa disebut pohon Huffman Sebagai contoh, dalam kode

ASCII string 7 huruf “ABACCDA” membutuhkan representasi 7 x 8 bit =56 bit (7 byte), dengan rincian sebagai berikut : Pada string di atas, frekuensi kemunculan A = 3, B = 1, C = 2, dan D = 1,\



Gambar 3.3 Pohon Huffman Untuk Karakter “ABACCDA”

3.3 Steganografi

3.3.1 Pengertian Steganografi

Menurut Kusniyati *et.al*, (2018) Steganografi berasal dari bahasa Yunani yaitu stegos yang berarti penyamaran dan graphia yang berarti tulisan. Steganografi digunakan untuk menyembunyikan informasi rahasia ke dalam suatu media sehingga keberadaan pesan

tersebut tidak diketahui oleh orang lain. Steganografi bertujuan untuk menghilangkan kecurigaan dengancara menyamarkan pesan tersebut.

3.3.2 Sejarah Steganografi

Menurut Nurhadian dan Pudoli, (2016) Steganografi berasal dari bahasa Yunani yang berarti tertutup atau tulisan ternsembunyi. Steganografi sudah dikenal sejak 440 SM. Herodutus menyebutkan salah satu contoh steganografi adalah Histiaeus mencukur kepada budak yang paling dipercayainya dan mentatokan sebuah pesan di atasnya. Setelah rambutnya tumbuh, kemudian budak tersebut diutus untuk membawa pesan rahasia di balik rambutnya.

3.3.3 Tujuan Steganografi

Tujuan dari steganografi adalah merahasiakan atau menyembunyikan keberadaan dari sebuah pesan tersembunyi atau sebuah informasi. Dalam prakteknya, kebanyakan pesan disembunyikan dengan membuat perubahan tipis terhadap data digital lain yang sisinya tidak akan menarik perhatian dari penyerang potensial, sebagai contoh sebuah gambar yang terlihat tidak berbahaya. Perubahan ini bergantung pada kunci (sama pada kriptografi). Dan pesan unutk disembunyikan orang yang menerima gambar kemudian dapat menyumpulkan informasi terselubung dengan cara menggati kunci yang benar ke dalam algoritma yang di gunakan. (Nurhadian dan Pudoli, 2016)

3.3.4 Steganografi EOF

Tujuan dari steganografi adalah merahasiakan atau menyembunyikan keberadaan dari sebuah pesan tersembunyi atau sebuah informasi. Pesan tersebut hanya disembunyikan ke dalam suatu media berupa gambar, teks, musik, atau media digital lainnya dan terlihat seperti pesan biasa. Teknik EOF atau End Of File merupakan salah satu teknik yang digunakan dalam steganografi. Teknik ini digunakan dengan cara menambahkan data atau pesan rahasia pada akhir dokumen. Teknik ini dapat digunakan untuk menambahkan data yang ukurannya sesuai dengan kebutuhan. Perhitungan kasar ukuran dokumen yang telah disisipkan data sama dengan ukuran dokumen sebelum disisipkan data ditambah ukuran data rahasia yang telah diubah menjadi encoding file. (Nurhadian dan Pudoli, 2016)

3.3.5 Metode End of File

Metode End of File (EOF) merupakan sa Implementasi Keamanan SMS Dengan Algoritma RSA Pada Smartphone Android lah satu teknik yang menyisipkan data pada akhir file. Teknik ini dapat digunakan untuk menyisipkan data yang ukurannya sama dengan ukuran file sebelum disisipkan kedalam file tersebut. Dalam teknik EoF, data yang disisipkan pada akhir file diberi tanda khusus sebagai pengenal start dari data tersebut dan pengenal akhir dari data tersebut.

3.4 Penelitian Terdahulu

Peneliti mengangkat beberapa jurnal sebagai referensi dan acuan yang digunakan dalam mengkaji penelitian yang dilakukan, seperti yang ditunjukkan oleh tabel 3.1

Tabel 3. 1 Penelitian Terdahulu

Nama Judul	Peneliti	Tahun dan ISSN	Hasil
Implementasi Keamanan SMS Dengan Algoritma RSA Pada Smartphone Android	Abdul, Riad, dan Hendra	Jurnal Teknologi Informasi & Komunikasi Tahun 2017, E-ISSN 2502-8332	pertukaran pesan dengan menggunakan fitur SMS pada zaman modern ini memiliki kelemahan yaitu pada faktor tingkat keamanan pesan, karena pesan dapat disadap oleh orang yang tidak berkepentingan pada saat pesan tersebut dikirim.
Nama Judul	**Penelitian**	**Tahun dan ISSN**	**Hasil**

Rancang Bangun Repository Guru Menggunakan Metode Enkripsi Advance Encryption Standard dan Kompresi Huffman (Studi Kasus: SMP Muhammadiyah 22 Tanggerang Selatan).	Abdul Dzuljalali	Jurnal Ilmiah Fifo Tahun 2019 ISSN: 2085-4315	Menurut hasil wawancara membuktikan bahwa proses yang berjalan sangatlah tidak efisien dan efektif serta rawan akan bocornya informasi penting dari dokumen tersebut ke pihak yang tidak bertanggung jawab
Kompresi File Menggunakan Algoritma Huffman Kanonik	Asrianda	Jurnal Penelitian Teknik Informatika Tahun 2017, ISSN: 2086-9479	Teknik kompresi algoritma Huffman mampu memberikan penghematan pemakaian memori sampai 30%. Algoritma Huffman mempunyai kompleksitas O(n log n) untuk himpunan dengan dan karakter

Nama Judul	Penelitian	Tahun dan ISSN	Hasil
Penerapan Metode Waterfall Pada Perancangan Sistem Informasi Pengolahan Data Nilai Siswa Sekolah Dasar.	Bariah	Perancangan Sistem dan Pengolahan Data Tahun 2020, ISSN: 2614-6606	Hasil pengujian sistem informasi yang dilakukan dengan metode b*lack* b*ox* dan berdasarkan nilai persentase dari hasil pengujian b*lack* b*ox* ini mendapatkan nilai persentase sebanyak 100% valid dan pengujian secara langsung oleh Guru Kelas dengan hasil cukup puas
Implementasi Algoritma RC4 Untuk Proteksi File MP3	Kirman	Jurnal Pseudocode Tahun 2018 ISSN: 2355-5920	Dari hasil pengujian yang berekstensi .mp3 enkripsi dan deskripsi berhasil dilakukan dengan memberikan informasi nama file, ukuran bytes, dan estimasi waktu.

Nama Judul	**Penelitian**	**Tahun dan ISSN**	**Hasil**
Penerapan Algoritma Rivert Code 4 (RC 4) Pada Aplikasi Kriptografi Dokumen	Kusniyati	Jurnal nasional Informatika dan teknologi jaringan Tahun 2018 ISSN: 2502-8332	Hasil yang akan dicapai dari penelitian ini adalah aplikasi kriptografi dokumen yang bias melakukan enkripsi dan dekripsi dokumen dengan algoritma Rivest Code 4 (RC 4).
Implementasi Keamanan File dengan Kompresi Huffman dan Kriptografi menggunakan AlgoritmaRC4 serta Steganografi menggunakan End of File Berbasis Desktop pada	Nurhadian	Teknologi Informatikadan Komunikasi Tahun 2016 ISSN: 2685-3310	Tingkat keamanan data soal setelah diembed cukup terjaga, dengan kata lain file tidak berkurang atau mengalami kerusakan setelah proses embed data dilakukan.

SMK Negeri 3 Kota Tangerang			

3.5 Ruang lingkup Penelitian

Ruang lingkup penelitian sangat dibutuhkan agar peneitian tetap terarah dan sesuai dengan tujuan dari penelitian ini. Ruang lingkup yang dimaksud adalah menentukan batasan batasan-batasan yang diperlukan untuk melakuan pengumpulan data sebagai bahan analisa data, perancangan sistem dan mendefinisikan kebutuhan-kebutuhan yang diperlukan.

Batasan-batasan penelitian didasarkan pada latar belakang serta tujuan penelitian. Adapun ruang lingkup dalam proses penelitian ini adalah peneliti memberikan layanan proses kompresi menggunakan algoritma huffman, dapat memberikan layanan enkripsi (pengacakan data) menggunakan algoritma RC4, dan dapat memberikan layanan proses penyisipan pesan rahasia kedalam gambar menggunakan algoritma steganografi, penyusuan tersebut diharapkan dapat bermanfaat untuk pengamanan data tersebut dari pihak lain.

BAB IV

METODE PENELITIAN

4.1. Lokasi dan Waktu Penelitian

4.1.1. Waktu Penelitian

Penelitian ini di mulai bulan maret sampai juli 2021 agar penelitian ini teararah dan tepat diselesaikan tepat waktu. Jadwal penelitian dapat dilihat pada tabel 4.1

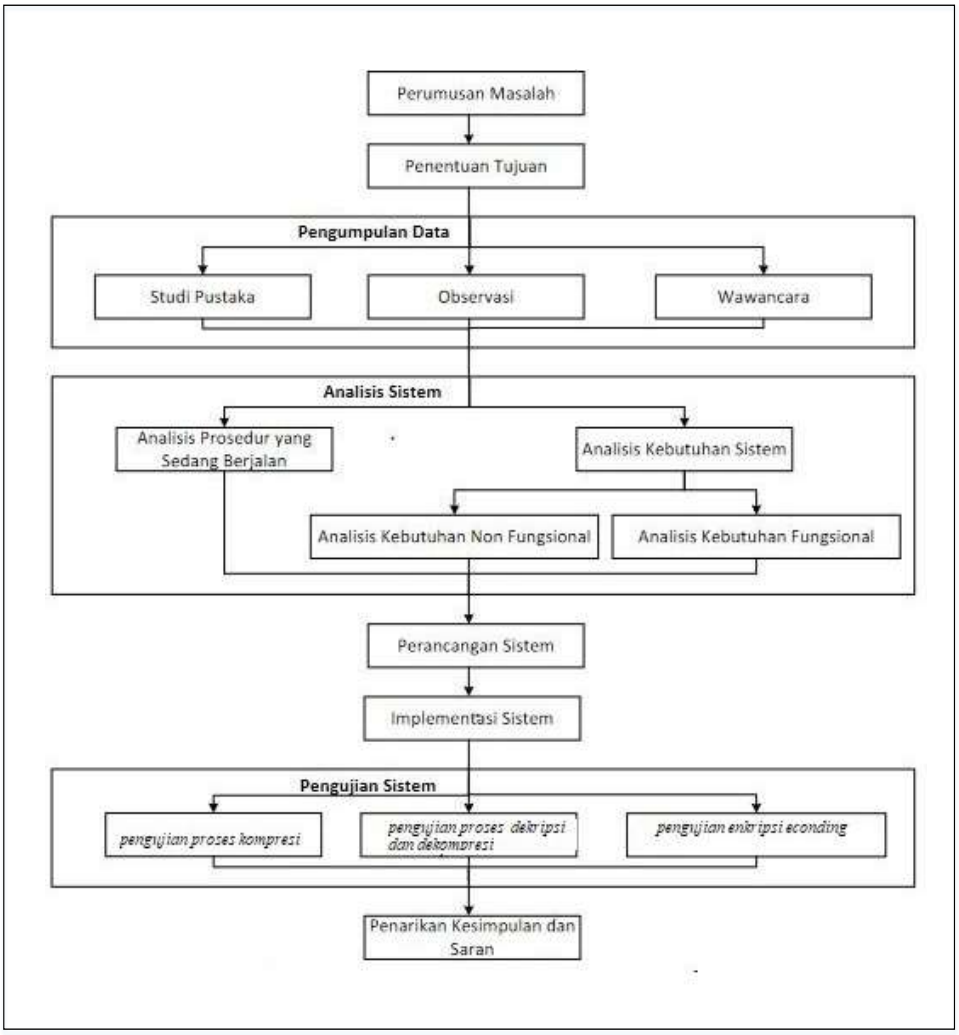
Tabel 4. 1 Waktu Penelitian

NO	Uraian Kegiatan	Tahun 2021																			
		Maret				April				Mei				Juni				Juli			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	Pemilihan Topik Penelitian																				
2	Pembuatan Proposal																				
3	Analisis Kebutuhan																				
4	Penerapan algortima huffam dan RC4																				
5	Perancangan																				
6	implementasi																				

4.2. Metode Penelitian Terapan

Dalam penelitian ini adalah metode penelitian terapan, yaitu penelitian yang berusaha mendeskripsikan suatu gejala, peristiwa, kejadian

yang terjadi pada saat sekarang. Secara keseluruhan kerangka kerja yang diterapkan adalah sebagai berikut, bisa di liat pada Gambar 4.1



Gambar 4.1 Metode Penelitian Terapan

1. Perumusan Masalah

Pada tahap ini dilakukan peninjauan ke sistem yang akan diteliti untuk mengamati serta melakukan eksplorasi lebih dalam dan menggali permasalahan yang ada pada sistem yang berjalan saat ini. Tahap perumusan masalah, merupakan langkah awal dari penelitian

ini, karena tahap ini diperlukan untuk mendefinisikan keinginan dari sistem yang tidak tercapai.

2. Penentuan Tujuan

Berdasarkan perumusan masalah yang telah dibuat pada tahap sebelumnya, maka tahap penentuan tujuan berguna untuk memperjelas kerangka tentang apa saja yang menjadi sasaran dari penelitian ini. Pada tahap ini ditentukan tujuan dari pembangunan sistem keamanan ini yaituuntuk meningkatkan keamanan data pada universitas sriwijaya palembang.

3. Pengumpulan Data

Pada tahap ini dilakukan pengumpulan data dan informasi untuk lebih mengetahui mengenai sistem yang diteliti. Dari data dan informasi yang dikumpulkan akan dapat diketahui mengenai sistem yang berjalan saat ini. Data-data dan informasi dapat diperoleh melalui wawancara dan pengamatan langsung. Adapun metode pengumpulan data yang dilakukan adalahdengan cara:

A. **Observasi** Yaitu penelitian langsung di universitas sriwijaya palembang untuk mengetahui prosedur kerja yang sedang berjalan.
B. **Wawancara** Pengumpulan data dengan proses tanya jawab langsung dan sistematis kepada narasumber Johansa, S.H. untuk mendapatkan data-data yang berkaitan dengan pengarsipan *file-file* maupun transaksi

database dari Divisi Data dan Infrastruktur kepada karyawan dan dosen.

4. Analisis Sistem

Pada analisis sistem, akan dilakukan analisis prosedur yang berjalan saat ini secara tidak langsung akan terlihat kelemahan-kelemahannya, sehingga saat itu juga bisa dilakukan analisa kebutuhan sistem yang bertujuan untuk mengidentifikasikan hal apa saja yang masih kurang dari prosedur sebelumnya untuk kemudian dilakukan langkah-langkah perbaikan. Analisis kebutuhan sistem di sini dibagi menjadi dua yaitu analisis kebutuhan fungsional terkait arsitektur sistem dan analisis kebutuhan non fungsional terkait kebutuhan perangkat keras, kebutuhan perangkat lunak maupun pengguna.

5. Perancangan Sistem

Perancangan sistem dilakukan guna mendapatkan gambaran dengan jelas tentang apa yang dikerjakan pada analisa sistem dan dilanjutkan dengan mempertimbangkan bagaimana membentuk sistem tersebut.

6. Implementasi Sistem

Pada tahap ini akan dilakukan penerapan/implementasi sistem yang mengacu pada perancangan sistem yang telah dibuat. Pengimplementasian sistem memiliki kriteriamudah digunakan dan dipahami oleh pemakai.

7. Pengujian Sistem.

   Pengujian sistem yang telah dibangun bertujuan guna mengetahui kesesuaian program dengan analisa sistem yang telah dibuat hingga dapat dipakai. Ada tiga pengujian yang akan dilakukan yaitu Pengujian proses kompresi dan enkripsi, Pengujian pengujian proses dekripsi dan dekompresi, dan pengujian penyisipan pesan rahasia kedalam gambar.

8. Penarikan Kesimpulan dan Saran

   Bagian ini berisi kesimpulan mengenai semua tahapan yang telah dilalui serta saran mengenai hasil dari penelitian yang telah dicapai.

4.3 Analisis dan Perancangan

4.3.1 Analisis System

Analisis system adalah teknik pemecahan masalah yang menguraikan bagian-bagian komponen dengan mempelajari seberapa baik bagian-bagian komponen tersebut bekerja dan berinteraksi dalam mencapai tujuan. Analisis sistem dilakukan untuk mengidentifikasi permasalahan-permasalahan yang ada sehingga sistem dapat berjalan sebagaimana mestinya. Pada penelitian ini terdapat dua analisis sistem yang dilakukan yaitu analisis masalah dan analisis kebutuhan.

4.3.2 Analisa Masalah

Dokumen soal merupakan data yang sangat penting bagi fakultas ilmu sosial dan ilmu politik Universitas Sriwijaya. Oleh karena

itu, sebuah dokumen seharusnya dijaga keasliannya dan kerahasiannya agar tidak disalahgunakan oleh orang yang tidak bertanggung jawab. Dikarenakan keamanan dokumen disini masih sangat kurang, sehingga terjadinya pencurian dokumen oleh orang

Yang tidak bertanggung jawab dan menyebarkannya. Salah satu cara untuk mengamankan sebuah dokumen yaitu dengan mengubah dokumen asli menjadi dokumen yang tidak bisa dibaca oleh orang lain atau sering disebut dengan enkripsi.

4.3.3 Analisis Kebutuhan

Dalam membangun sebuah sistem, perlu dilakukan tahap analisis kebutuhan sistem untuk mengetahui kebutuhan yang diperlukan oleh sistem secara menyeluruh. Analisis kebutuhan sistem dapat dikelompokkan menjadi 2 bagian yaitu kebutuhan fungsional dan kebutuhan non- fungsional.

4.6.1. Kebutuhan Fungsional

Kebutuhan fungsional yang harus dipenuhi dari sistem yang dirancang adalah sebagai berikut:

a) Sistem dapat melakukan enkripsi data terhadap file teks dengan menggunakanalgoritma RC4.
b) Sistem kompresi data per bit hasil dari (prefix code) dengan menggunakan akar daundari algoritma huffam.

c) Sistem dapat memverifikasi keutuhan file yang diterima dengan mencocokkan nilaihash yang ada.

d) Sistem dapat mengembalikan pesan atau file ke bentuk semula, yaitu dengan tahapandekompresi untuk proses pengembalian *file* ke bentuk dan ukuran yang semula.

4.6.2. Kebutuhan Non-Fungsional

Kebutuhan non-fungsional adalah kebutuhan yang berisi properti perilaku yang dimiliki oleh sistem. Kebutuhan non-fungsional dari sistem yang dirancang adalah sebagai berikut:

e) **Kinerja**, Kebutuhan non-fungsional adalah kebutuhan yang berisi properti perilaku yang dimiliki oleh sistem (Fatta, 2007). Kebutuhan non-fungsional dari sistem yang dirancang adalah sebagai berikut:

f) **Informasi,** Sistem yang di bangun mampu mengenkripsi dan mendekripsi data secaracepat

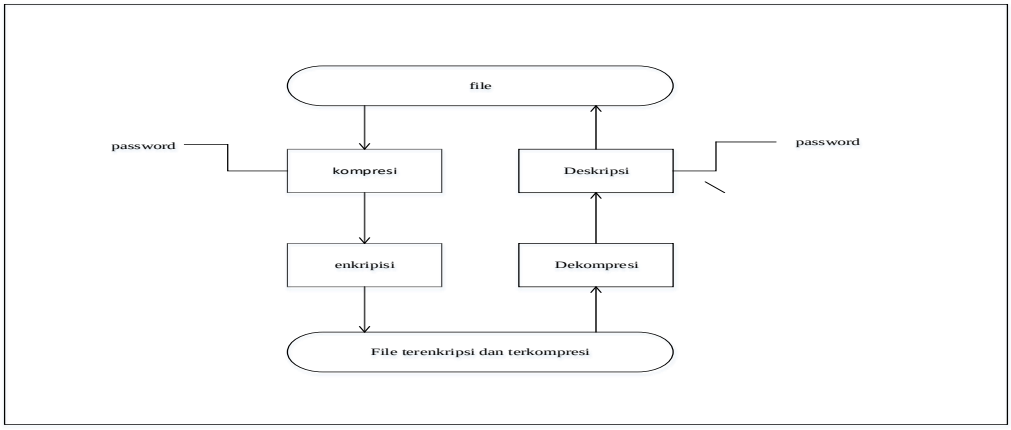
g) **Mudah diplajari dan digunakan,** Sistem yang dibangun bersifat user friendly sehingga sistem dapat dengan mudah dipelajari dan digunakan oleh pengguna.

h) **Hemat biaya**, Sistem yang dibangun tidak memerlukan perangkat tambahan maupun sistem pendukung dalam penggunaannya.

4.3.2 Perancangan Sistem

Secara umum, rancangan program yang akandibuat dapat

dilihat pada gambar 4.2



Gambar 4.2 Perancangan Sistem

4.4. Alat dan Teknik Pengembangan Sistem

4.4.1. Data Flow Diagram (DFD)

Menurut Sukamto ‘dkk’ (2014:288), “Data Flow Diagram atau dalam bahasa Indonesia menjadi Diagram Alir Data (DAD) adalah refresentasi grafik yang menggambarkan aliran informasi dan transFormasi informasi yang diaplikasikan sebagai data yang mengatur dari masukan (Input) dan keluaran (Output). DFD tidak sesuai untuk memodelkan sistem yang menggunakan pemograman berorientasi objek.” Menurut sukamto ‘dkk’ (2014:288), notasi-notasi DFD dilihat pada tabel 4.2

Tabel 4.2 Simbol-Simbol Data Flow Diagram

No	Simbol	Nama	Keterangan
1		Proses Transformasi	Proses yang mengubah data dari input menjadi output
2		Sumber & Tujuan Data	Karyawan & organisasi yang mengirim data ke dan menerima data dari sistem.
3		Arus Data	Arus data yang masuk ke dalam dan keluar dari sebuah proses.
4		Penyimpanan Data	Penyimpanan Data

Sumber : Sukamto dan Shalahuddin (2014:71)

Menurut Sukamto dan Shalahuddin (2014:72), berikut ini adalah stahapan-tahapan perancangan menggunakan *DFD*:

a. Membuat *DFD Level* 0

   Context Diagram DFD Level 0 menggambarkan sistem yang akan dibuat sebagai suatu entitas tunggal yang berinteraksi dengan orang maupun sistem lain. *DFD Level* 0 digunakan untuk menggambarkan interaksi antara sistem entitas luar.

b. Membuat *DFD Level* 1

   DFD Level 1 digunakan untuk menggambarkan modul-modul yang ada dalam sistem yang aakan dikembangkan. *DFD Level* 1 merupakan hasil *breakdown DFD Level* 0 yang sebelumnya sudah dibuat.

c. Membuat *DFD Level* 2

Modul-modul pada *DFD Level* 1 dapat di *breakdown* menjadi *DFD Level* 2. Modul mana saja yang harus di *breakdown* lebih detail tergantung pada tingkat kedetilan modul tersebut. Apabila modul tersebut sudah cukup detail dan rinci maka modul tersebut sudah tidak perlu untuk di *breakdown* lagi. Untuk jumlah *DFD Level* 2 sama dengan jumlah modul pada *DFD Level* 1 yang di *breakdown*.

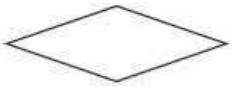
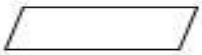
d. Membuat *DFD Level* 3

DFD Level 3, 4, 5 dan seterusnya merupakan *breakdown* dari modul pada *DFD Level* di atasnya. *Breakdown* pada *level* 3, 4 dan 5 dan seterusnya aturannya sama persis dengan *DFD Level* 1 atau *Level* 2.

4.4.2. *Entity Relationship Diagram* (*ERD*)

Menurut Sukamto ‘dkk’ (2014:50-289), “*Entitiy Relationship Diagram* (*ERD*) adalah pemodelan awal basis data yang akan dikembangkan berdasarkan teori himpunan dalam bidang matematika untuk pemodelan basis data relasional”. *ERD* memiliki beberapa aliran notasi seperti notasi Chen (dikembangkan oleh Peter Chen). Barker (dikembangkan oleh Richard Barker, Ian Palmer, Harry Ellis), notasi Crow’s Foot, dan beberapa notasi lain. Namun yang banyak digunakan adalah notasi dari Chen. Berikut adalah simbol-simbol yang digunakan pada *ERD* dengan notasi Chen dilihat pada tabel 4.3

Tabel 4. 3 *Entity Relationship Diagram (ERD)*

Notasi	Keterangan
	Entitas, yaitu kumpulan dari objek yang dapat diidentifikasi secara unik.
	Relasi, yaitu hubungan yang terjadi antara satu atau lebih entitas. Jenis hubungan antara lain: satu ke satu, satu ke banyak, dan banyak ke banyak.
	Atribut, yaitu karakteristik dari entity atau relasi yang merupakan penjelasan detail tentang entitas.
	Garis, hubungan antara entity dengan atribut nya dan himpunan entitas dengan himpunan relasi.
	Input/output data, yaitu proses input/output data, parameter, informasi.

Sumber : Sukamto dan Shalahuddin (2014:50-289)

4.4.2 Flowchart

Menurut Indrajani (2015:36-38), "*Flowchart* adalah penggambaran secara grafik dari langkah-langkah dan urutan prosedur suatu program." menjelaskan simbol-simbol dalam *Flowchart* dapat dilihat pada tabel 4.4

Tabel 4.4 *Flowchart Diagram*

Simbol	Kegunaan
	menghubungkan antara simbol yang satu dengan simbol yang lainnya.
	keluar/masuk prosedur atau proses dalam lembar/halaman yang lain.
	keluar/masuk proses dalam lembar/halaman yang sama.
	menunjukkan pengolahan yang dilakukan oleh komputer.
	menunjukkan pengolahan yang tidak dilakukan oleh komputer.
	kondisi yang akan menghasilkan beberapa kemungkinan jawaban/aksi.
	permulaan atau akhir dari suatu program.
	menunjukkan bahwa data di dalam simbol ini akan disimpan secara sementara.
	menunjukkan bahwa data di dalam simbol ini akan disimpan secara permanen.
	proses input dan output tanpa tergantung dengan jenis peralatannya.
	input berasal dari dokumen dalam bentuk kertas atau output dicetak ke kertas.

Sumber : Indrajani (2015:36)

4.5. Perancangan Sistem

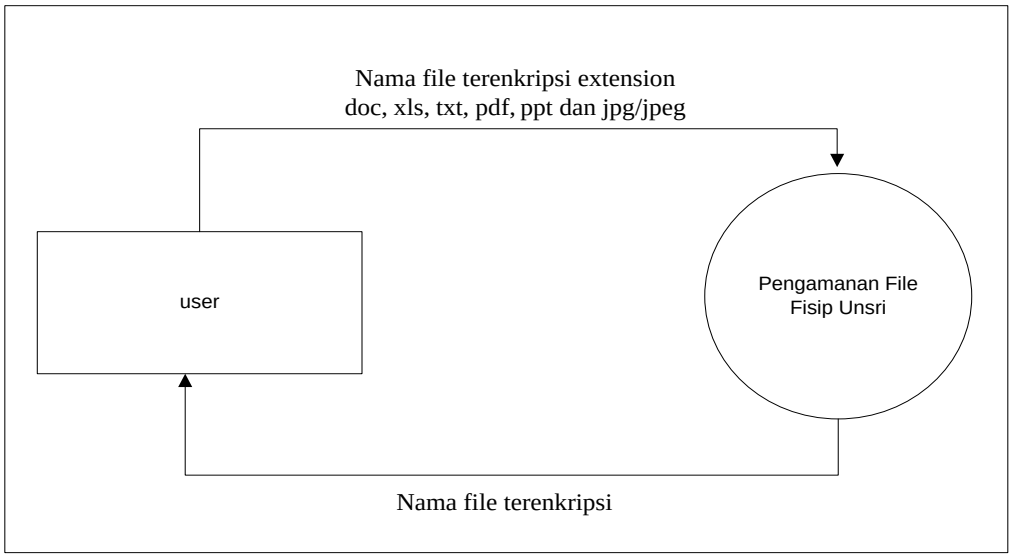
Rancangan system secara umum dilakukan dengan maksud untuk memberikan gambaran tentang system yang akan diusulkan. Rancangan ini mengindentifikasi komponen komponen system informasi yang akan dirancang secara rinci. Adapun rancangan system ini adalah sebagai berikut :

4.5.1. Perancangan DFD Pengamanan File Fisip Unsri

Data flow diagram (DFD) merupakan cara atau metode untuk membuat rancangan sebuah sistem yang berorientasi pada alur yang bergerak pada sebuah sistem nanti yang akan dibuat. Dalam pembuatan sistem informasi DFD sering digunakan. DFD dibuat oleh para analis untuk membuat sebuah sistem yang baik dimana DFD ini akan diberikan kepada para programer untuk melakukan proses coding program aplikasi sistem inform.Dalam aplikasi pengamanan file fisip unsri, DFD yang dipakai,yaitu

1. DFD *Diagram Konteks* Aplikasi Pengamanan File Fisip Usnsri

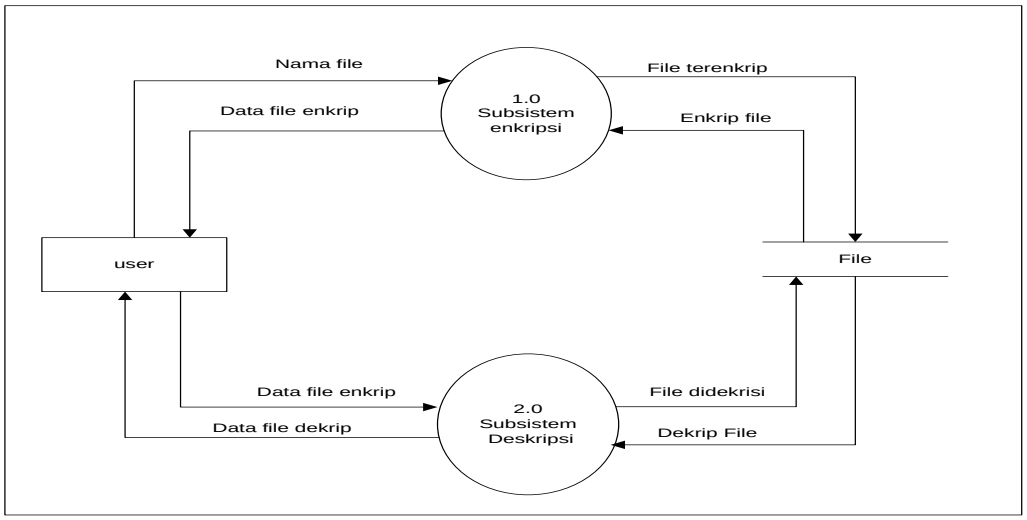
Diagram konteks pada gambar 4.3 menjelaskan aliran data pada sistem kriptografi yang dibangun. Aplikasi yang akan dibangun memiliki sebuah terminator, yaitu pengguna sistem. Aliran data yang masuk ke sistem berupa nama *file* yang extension di izinkan terenkripsi berupa file yang berbentuk doc, xls, txt, pdf, ppt dan jpg atau jpeg. Aliran data yang dihasilkan oleh sistem berupa *file* extension terenkripsi.



Gambar 4.3 *DFD Diagram Konteks*

2. DFD *Level* 0 Aplikasi Pengamanan File Fisip Unsri

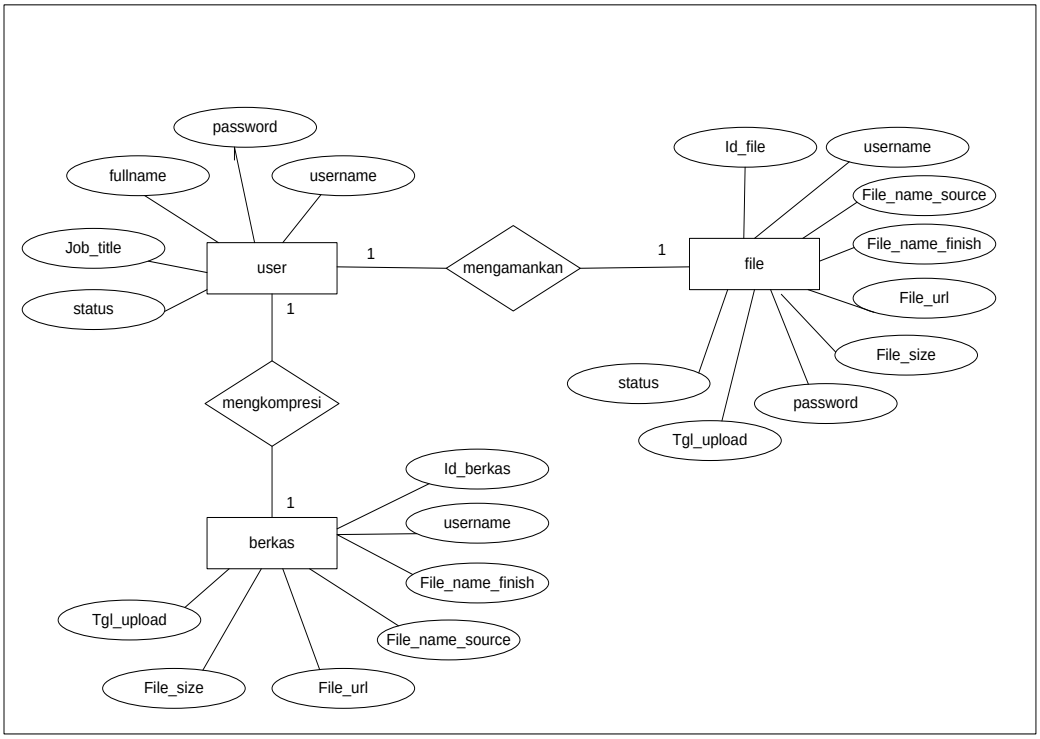
Pada gambar 4.4 menunjukkan pembagian sistem kriptografi yang dibangun menjadi dua subsistem, yaitu subsistem enkripsi dan subsistem dekripsi.



Gambar 4.4 DFD Level 0

3. Perancangan *ERD* Pengamanan File Fisip Unsri

ERD menjelaskan objek data, atribut, keterhubungan, dan berbagai jenis indikator pada aplikasi yang dibangun dan siapa saja yang berinteraksi dengan sistem. Pada gambar 4.5 dapat dilihat ada tiga entitas / entity yaitu data user, data file dan data berkas. Atribut data user mempunyai lima atribut yaitu username, password, fullname, job_title dan status. sedangkan atribut data file sepuluh atribut yaitu id_file, username, file_name_source, file_name_finish, file_url, file_size, password, tanggal_upload, status dan keterangan. Ke dua atribut tersebut mempunyai hubungan yaitu satu pengguna memiliki satu file.



Gambar 4. 5 *ERD* Pengamanan File Fisip Unsri

4.5.2. Perancangan Struktur Database

Database terdiri dari beberapa table yang digunakan untuk menyimpan record-record pada aplikasi pengamanan fisip unsri. Adapun spesifikasi data dari *database* aplikasi pengamanan file sebagai berikut,yaitu:

1. Tabel User

User adalah tabel data user digunakan untuk menyimpan data user sebagai pengelola yang berisi username, password, fullname, status dan job_title. Adapun tabel user dapat terlihat pada Tabel 4.5

Tabel 4.5 User

No.	***Field***	***Type***	***Size***	**Keterangan**
1	username	varchar	15	**Primary Key*
2	password	varchar	10	Password user
3	fullname	varchar	50	Nama lengkap
4	Job_title	varchar	50	Jabatan

2. Tabel File

data file adalah tabel yang digunakan untuk menyimpan data file yang terenkrip maupun dienkripis yang berisi id_file, username, file_name_source, file_name_finish, file_url, file_size, tanggal_ulpload, status, dan keterangan. Adapun tabel data file dapat terlihat pada Tabel 4.6

Tabel 4.6 Data File

No.	***Field***	***Type***	***Size***	**Keterangan**
1	id_file	Int	11	id **Primary Key*
2	Username	Varchar	15	Username
3	file_name_source	Varchar	256	nama file
4	file_name_finish	Varchar	255	nama file akhir
5	file_url	Varchar	255	
6	file_size	Float		
7	Password	Varchar	16	
8	tgl_upload	timestamp		
9	Status	Enum		
10	keterangan	varchar	255	

3. Tabel File

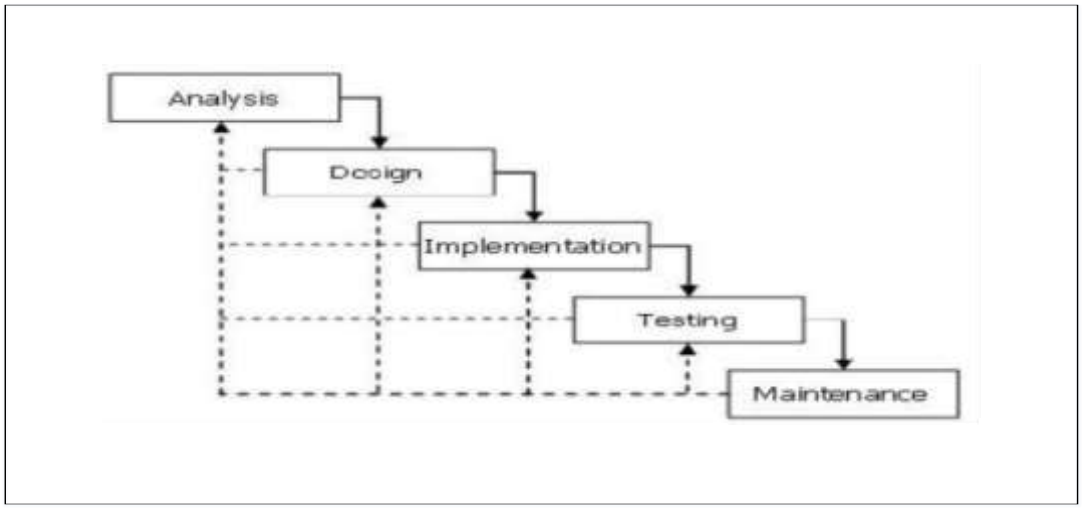
data berkas adalah tabel yang digunakan untuk menyimpan data file yang terenkrip maupun dienkripis yang berisi id_berkas, username, file_name_source, file_name_finish, file_url, file_size, tanggal_ulpload, status, dan keterangan. Adapun tabel data file dapat terlihat pada Tabel 4.6.

Tabel 4.7 Data Berkas

No.	***Field***	***Type***	***Size***	**Keterangan**
1	id_berkas	Int	11	id **Primary Key*
2	Username	Varchar	15	Username
3	file_name_source	Varchar	256	nama file
4	file_name_finish	Varchar	255	nama file akhir
5	file_url	Varchar	255	
6	file_size	Float		
7	tgl_upload	timestamp		
8	status	Enum		
9	keterangan	varchar	255	

4.6 Teknik Pengembangan System Waterfall

Menurut Bariah & Putera, (2020) mengungkapkan bahwa Dalam hal pengembangan serta perencanaan sistem perangkat lunak penulis menggunakan metode pengembangan perangkat lunak yang bertujuan untuk melakukan enkripsi supaya terhindar dari pencurian data sehingga menjaga keamanan data yang dikirim pengguna dan penerima tetap aman dalam model air terjun (waterfall). Berikut adalah bentuk diagram model waterfall beserta penjelasannya. dilihat pada gambar 4.3.



Gambar 4.6 Diagram Model Waterfall

4.6.1 Analysis

Merupakan proses pengumpulan kebutuhan perangkat lunak. Untuk memahami dasar dari program yang akan dibuat, seorang analisis harus mengetahui ruang lingkup informasi, fungsi- fungsi yang dibutuhkan, kemapuan kinerja yang ingin dihasilkan dan perancangan antarmuka pemakai perangkat lunak tersebut.

Pada fase ini penulis malakukan pemahamam serta analisis terhadap data yang telah didapatkan pada saat wawancara sebelumnya. Komponen fase ini meliputi:

a. Indentifikasi masalah, permasalahan bisa teridentifikasikan setelah melaukan wawancara dengan pihak Fakultas Ilmu Sosial Dan Ilmu Politik Universitas Sriwijaya dengan Narasumber Johansa, S.H.
b. Kebutuhan system, pada fase ini menganalisis dari hasil wawancara, maka ditentukan spesifikasi untuk menerapkan aplikasi keaman data.

c. *Planning,* Pada fase ini penelitian menentukan estimasi waktu pengembangan system dan menentukan urutan langkah pengembanganya.

4.6.2 Design

Spesifikasi kebutuhan dari tahap sebelumnya akan dipelajari dalam fase ini dan desain sistem disiapkan. Desain Sistem membantu dalam menentukan perangkat keras(hardware) dan sistem persyaratan dan juga membantu dalam mendefinisikan arsitektur sistem secara keseluruhan.

Pada fase ini penulis menerjemakan semua hasil dari fase analisis menjadi representase dari perangkat lunak unutk fase berikutnya, berikut komponen design:

a. Merancang diagram-diagram yang menggambarkan struktur dasar yang berkaitan dengansystem keaman data.
b. *Database* design, merancang struktur *database* yang terdiri dari kumpulan table yangmemiliki kolom yang saling berelasi antara masing masing table.

4.6.3. Implementasi

Pada tahap ini, sistem pertama kali dikembangkan di program kecil yang disebut unit, yang terintegrasi dalam tahap selanjutnya. Setiap unit dikembangkan dan diuji untuk fungsionalitas yang disebut sebagai unit *testing*.

Pada tahap ini penulis melakukan beberapa fase implementasi, diantaranya:

a. Mengimplementasikan Enkripsi

   Pada tahap ini penulis menggunakan algoritma RC4 dan algoritma steganografi pada setiap file yang akan dienkripsi.

b. Mengimplementasikan Kompresi Huffman

   Pada tahap ini penulis menggunakan kompresi Huffman pada setiap file yang akan di- upload pada saat memasukan file ke aplikasi.

4.6.4. Testing

Proses ini akan menguji kode program yang telah dibuat dengan memfokuskan pada bagian dalam perangkat lunak. Tujuanya untuk memastikan bahwa semua penyataan telah diuji dan memastikan juga bahwa imput yang digunakan akan menghasilkan output yang sesuai. Pada tahap ini pengujian dibagi menjadi dua bagian, pengujian internal dan pengujian eksternal. Pengujian internal bertujuan menggambarkan bahwa semua statement sudah dilakukan pengujian, sedangkan pengujian eksternal bertujuan untuk menentukan kesalahan serta memastikan output yang dihasilan sesuai dengan yang diharapkan.

Pada tahapan ini penulis malakukan fase testing menggunakan blac box, diantaranya:

Pengujian black box pada perangkat lunak bertujuan menemukan kesalahan dalam ketegori:

3.1.1 Fungsi – fungsi yang tidak benar atau hilang

3.1.2 Kesalahan *interface*

3.1.3 Kesalahan dalam struktur data atau akses *database* ekternal.

3.1.4 Kesalahan kinerja

3.1.5 Inisialisasi dan kesalahan terminasi

4.6.5 Maintenance

Semua tindakan teknik dan administratif yang dilakukan untuk menjaga agar kondisi mesin/peralatan tetap baik dan dan dapat melakukan segala fungsinya dengan baik, efisien, dan ekonomis sesuai dengan tingkat keamanan yang tinggi. Pada tahap ini penulis melakukan perawatan secara berkala dengan perawatan yang tepat. Terdapat dua hasil yang diharapkan dari kegiatan perawatan, yaitu:

a) Condition maintenance, yaitu aktivitas perawatan untuk mempertahankan keadaan mesin/peralatan agar dapat berfungsi dengan baik sesuai dengan usia ekonomis mesin itu.
b) Replacement maintenance, yaitu aktivitas perawatan untuk perbaikan dan penggantian komponen mesin tepat pada waktunya sesuai dengan jadwal yang telah direncanakan.

4.7 Pengujian

Pengujian sistem merupakan tahap mengidentifikasi hasil dari

implementasi sistem apakah sistem telah berjalan sesuai dengan fungsi-fungsi yang sebelumnya ditentukan pada tahapanalisis dan perancangan system.

Pengujian dilakukan sebagai tlak ukur keberhasilan dalam penelitian untuk mengetahui kelebihan serta kekurangan pada sebuah penelitian tersebut.

4.7.1 Aspek Mutu (*Fidelity*)

Mutu (*Fidelity*), pengujian dalam hal ini yaitu memeriksa kualitas citra terhadap manipulasi untuk mengetahui kemiripan citra sebelum dan sesudah proses penyisipan data.

1. **MSE (*Mean Square Error*)**

MSE adalah ukuran yang digunakan untuk menilai seberapa baik sebuah metode dalam melakukan rekunstruksi atau restorasi citra relative terhadap citra aslinya.

$$MSE = 1 \quad MxN \quad \sum M\ x \sum N\ y\ [f1(x, y) - f2(x, y)]\ 2$$

Semakin kecil nilai MSE, maka hasil pemrosesan citra semakinmendekati citra aslinya.

2. **PSNR (*Peak Signal to Noise Ratio*)**

PSNR adalah ukuran perbandingan nilai maksimum dari kedalaman bit citra yang diukur dengan besarnya *noise* yang berpengaruh pada sinyal tersebut. Besarnya *noise* diwakili oleh nilai MSE.

$$\text{PS} = 20 \quad Log_{10} \quad \left(\frac{255}{\sqrt{MSE}}\right)$$

Semakin besar nilai PSNR, maka hasil pemrosesan citra semakin mendekati citra aslinya. Bisa di lihat pada tabel 4.7 kualitas citra nilai dari PSNR tersebut:

Tabel 4.7 Nilai PSNR

Nilai PSNR	Kualitas Citra
60 dB	*Excellent*, tanpa derau
50 dB	*Good*, terdapat sejumlah derau, tetapi kualitas citra masih bagus
40 dB	*Reasonable*, terdapat butiran halus atau seperti salju di dalam citra
30 dB	*Poor picture*, terdapat banyak derau
20 dB	*Unusable*

Sumber: (Barovih,et.al,. 2021)

4.7.2 Tingkat ketahanan (*Robustness*)

Pengujian dilakukan beberapa uji serangan yang bertujuan untuk mengetahui tingkat ketahanan citra terhadap serangan luar. Uji serangan yang dilakukan yaitu diantaranya kecerahan, ketajaman, rotasi dan *resize*. Pengujian citra hasil ditunjukkan oleh Tabel 4.8.

Tabel 4.8 Pengujian Serangan Citra Hasil

Operasi Dasar	*Nilai Perubahan*	*Target Hasil Uji*	*Target PSNR(dB)*
Kecerahan	-5 sampai +25	*Success/Failed*	>= 40dB
Ketajaman	-5 sampai +25	*Success/Failed*	>= 40dB
Rotasi	90°/180°/270°	*Success/Failed*	>= 40dB
Resize	Diperbesar 2x	*Success/Failed*	>= 40dB

Sumber: (Barovih,et.al,. 2021)

Pengujian menggunakan 4 sampel citra 24-bit, yaitu citra jenis bmp, jpeg, png dan tiff dengan dimensi yang sama. Citra hasil uji serangan dikatakan berhasil jika pesan yang tersisip pada citra hasil tidak mengalami kerusakan setelah meng-*decode* menggunakan metode MLSB.

4.7.3 Pengujian Terhadap Perbandingan Isi Pesan

Pengujian dilakukan dengan menjaga keutuhan dan ukuran pesan sebelum dan sesudah proses steganografi. Pengujian ini dapat dikatakan berhasil jika pesan asli dan pesan hasil *decode* sama persis baik isi maupun ukuran pesannya.

4.7.4 Pengujian Black Box

Pengujian system menggunakan Teknik pengujian *Black Box testing.* Pengujian ini merupakan memperoleh kondisi Input seluru keperluan fungsional program.

Tabel 4.9 Pengujian Black Box

No	*Form* yang diuji	Keterangan	Harapan	Hasil
1	Pengujian pada *Form login*	Pada *Form login* pengguna harus memasukan *username* dan *password* kemudianklik tombol *login.*	Pengujian pada saat*login* "sukes" anda masuk ke menu utama. Jika salah maka kembali ke menu logindan kembali memasukan *username*dan *passwaord.*	**Berhasil**
2	Pengujian pada *Form* kirim . upload *file*	Pada saat pengguna mengirim/mengu pload sebuah file, kemudian file tersebut di ekripsi.lalu masuk ke folder enkrip.	Ketika file telah di enkripsi lalu *file* terenkripsi masuk ke folder enkrip dengan data acak.*file* tersebut berarti berhasil	**Berhasil**

3	Pengujian pada kontak masuk daftar list	Pada *Form* kontakmasuk pengguna dapat melihat *file* yang telah dikirim. Baik di database dan di aplikasi itu sendiri.	Pengujian pada saat pengguna melihat kontak masuk maka akan menampilkan *file* yang telah dikirim oleh pengguna lain	**Berhasil**
4	Simuasi	Pada *Form* ini dilakukan simulasi untuk melihat kunci kunci private, digital hasil dari ekripsi dan hasil dekripsi	Pada *Form* ini dilakukan simulasi untuk melihat kunci kunci private, digital hasil dari ekripsi dan hasil dekripsi	**Berhasil**
6	Pengujian kompresi	File yang di kompresi apakah berkuranng/bertambah dari file sebelumnya	File yang di kompresi berkurang/bertambah saat di kompresi	**Berhasil**

7	Pengujian penysisipan pesan kedalam gambar	Pada saat pengguna mengirim/meng upload sebuah file, kemudian file tersebut di ekripsi gambar lalu masuk ke folder enkrip.	Ketika file telah di enkripsi gambar lalu *file* terenkripsi masuk ke folder enkrip dalam bentuk format gambar, file tersebut berarti berhasil	**Berhasil**

4.7.5 Pengujian White Box

Salah satu teknik pengujian menggunakan sistem *WhiteBox Testing* adalah *Basis Path Testing*. Metode *Basis Path* digunakan untuk menentukan ukuran kompleksitas logika dari suatu logika. Metode *Basis Path Testing* yang digunakan berguna untuk:

a. Mengukur kompleksitas *logic* dari desain prosedur dan sekaligus sebagai pedoman untuk mendapatkan konsistensi jalur aplikasi.

b. Pengujian yang dilakukan dijamin menggunakan statement dalam program minimal satu kali selama pengujian.

c. Menghitung *cylometris complexity* sebagai ukuran kontitif untuk menentukan jumlah *independent* path sebagai jalur yang

perlu diuji.

Notasi grafik alir (*flowgraph*) adalah grafik program yang dihasilkan dari pemetaan *flowchart* program yang ada untuk merepresentasikan aliran kontrol logika program yang ada. Dalam notasi grafik alir (*flowgraph*) dikenal beberapa istilah, yaitu :

a. Jumlah region (R) grafik alir (*flowgraph*) sama dengan kompleksitas siklomatis
b. Kompleksitas Siklomatis, V(G), untuk grafik alir G ditentukan sebagai V(G) = (E – N) + 2, dimana E adalah jumlah *edge* grafik alir dan N adalah jumlah simpul grafik alir.
c. Kompleksitas Siklomatis, V(G), untuk grafik alir G ditentukan sebagai V(G) = P + 1, dimana P adalah jumlah simpul predikat yang diisikan dalam grafik alir.

Rumus pengujian basis path : V(G) = Σ R , V(G) = E – N + 2 atau V(G) = P + 1.

Pengujian White Box pada aplikasi ini menggunakan Cyclomatic Complexity, pada konteks metode Basis Path Testing, nilai yang dihitung bagi Cyclomatic Complexity menentukan jumlah jalur – jalur yang independen dalam kumpulan basis suatu program dan memberikan jumlah tes minimal yang harus dilakukan untuk memastikan

bahwa semua pernyataan telah di eksekusi sekurangnya satu kali. Jalur independen adalah tiap jalur pada program yang memperlihatkan satu kelompok baru dari pernyataan proses atau kondisi baru. Teknik White Box Testing adalah sebagai berikut:

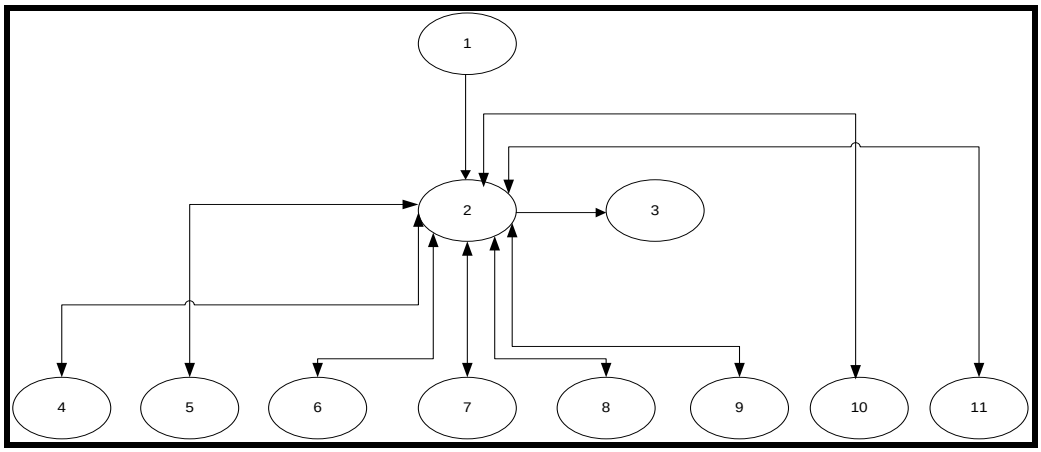
a. Basis Path Testing
b. Buat Flow Graph Notation
c. Hitung Cyclomatic Complexity V(G) = E – N + 2

   Keterangan:

   E = Jumlah busur pada flow graph

   N = Jumlah simpul pada flow graph.
d. Tentukan jalur bebas (independent path) = jalur program yang merupakan satu kumpulan perintah pengolahan atau satu kondisi pengolahan.
e. Siapkan kasus uji untuk setiap jalur bebas.
f. Graph Matrices = Connection Matrices = representasi lain dari Flow Graph Notation

Berikut ini adalah gambar dari Pengujian White Box :



Gambar 4.7 White Box Testing

Keterangan Gambar :

1. Login
2. Dashboard
3. Log Out
4. Kompresi
5. Dekompresi
6. Enkripsi RC4
7. Dekripsi RC4
8. Enkripsi STG-EOF
9. Dekripsi STG-EOF
10. Daftar List
11. Bantuan

Path yang dapat dibentuk adalah sebagai berikut :

1. 1-2-4-3
2. 1-2-5-3
3. 1-2-6-3

4. 1-2-7-3
5. 1-2-8-3
6. 1-2-9-3
7. 1-2-10-3
8. 1-2-11-3
9. 1-2-3

Terdapat 9 Path yang dibentuk dari diagram alir diatas, selanjutnya untuk memastikan dengan benar jumlah Path dalam satu diagram alir dapat digunakan rumus yaitu :

$V(G) = E - N + 2$

$= 18 - 11 + 2$

$= 9$

Dari Path yang didapat melalui diagram alir dan perhitungan dengan menggunakan rumus diperoleh hasil yang sama, dengan demikian berarti sistem sudah berjalan dengan benardan selanjutnya akan dilalukan pengujian.

BAB V

HASIL DAN PEMBAHASAN

5.1. Hasil

Dalam pembuatan perancangan aplikasi keamanan data dokumen ini menggunakan metode *waterfall*, adapun tahapannya adalah sebagai berikut:

5.1.1. Analisis

5.1.1.1. Identifikasi Masalah

Identifikasi permasalahan yang terjadi pada rancangan aplikasi keamanan data dokumen dapat dilihat pada tabel 5.1

Tabel 5. 1 Identifikasi Masalah

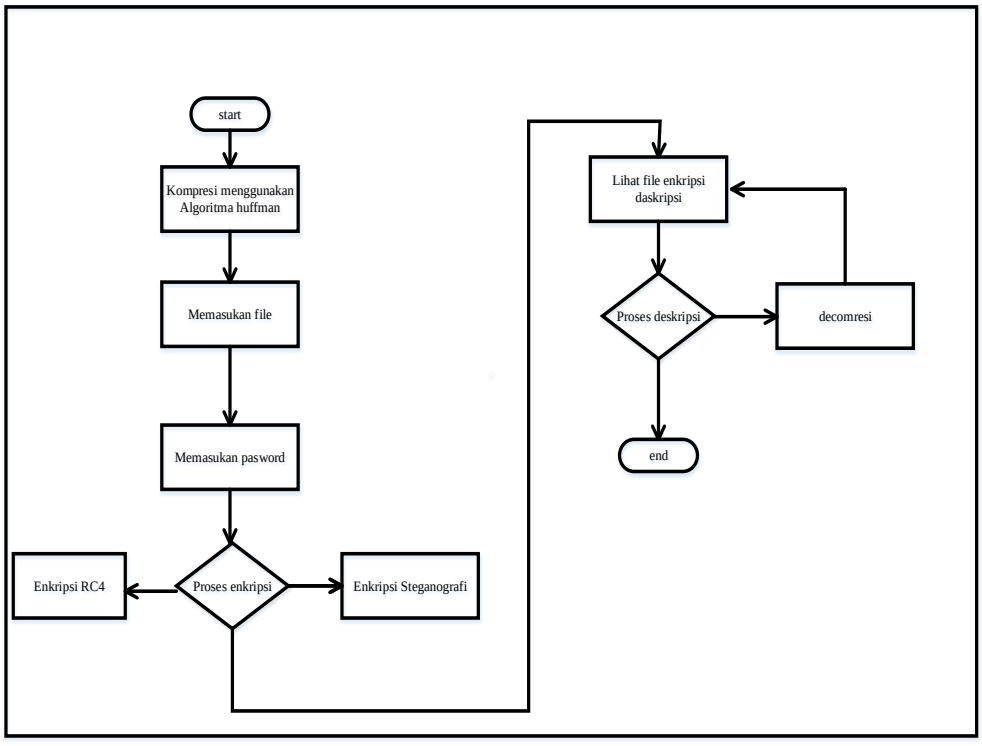
No	Masalah	Penyebab Masalah
1	Kerusakan Data	Pada saat pengiriman data dapat terjadi kesalahan penanda tanganan *file* atau pengesahan yang tidak cocok sehingga *file* tersebut tidak dapat ditampilkan
2	Perubahan Data	Pada saat pengiriman data dapat terjadi pembacaan data oleh orang lain ketika data sedang dikirim sehingga dapat terjadi perubahan data *file* aslinya.
3	Pencurian Data	Pada saat pengiriman data dapat terjadi pengambilan data oleh orang lain.

5.1.1.2 Alur Sistem Yang Diusulkan

Pada gambar 5.1 *Flowchart* sistem yang diusulkan terdapat 3 proses utama, yaitu proses kompresi, enkripsi *file* dan deskripsi *file.*

a. Proses kompresi, enkripsi dan deskripsi

yang digunakan untuk menelusuri proses program pada aplikasi untuk keamanan file. Flowchart ini merupakan alur jalannya proses enkripsi sebuah file yang ingin dienkripsi. Flowchart proses enkripsi dapat dilihat seperti Gambar 5.1

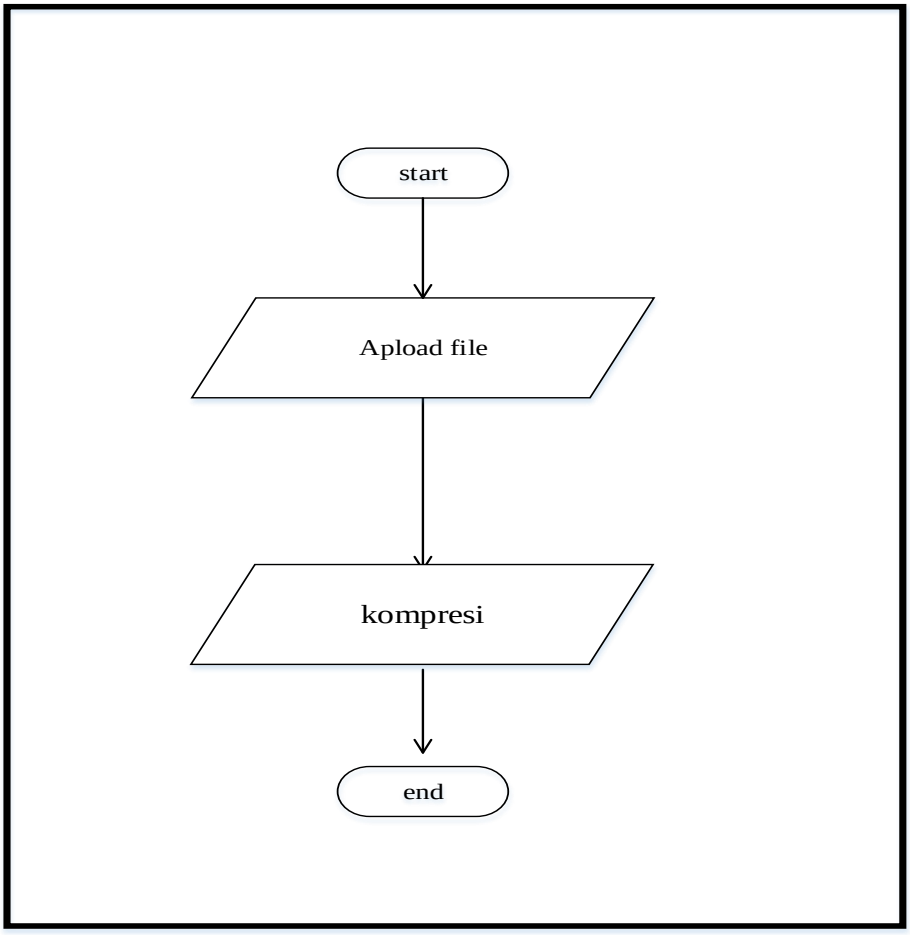


Gambar 5.1 Flowchart sistem yang diusulkan

Pada gambar 5.1 *Flowchart* sistem yang diusulkan terdapat 3 proses utama, yaitu proses kompresi, enkripsi dan deskripsi.

a. Kompresi

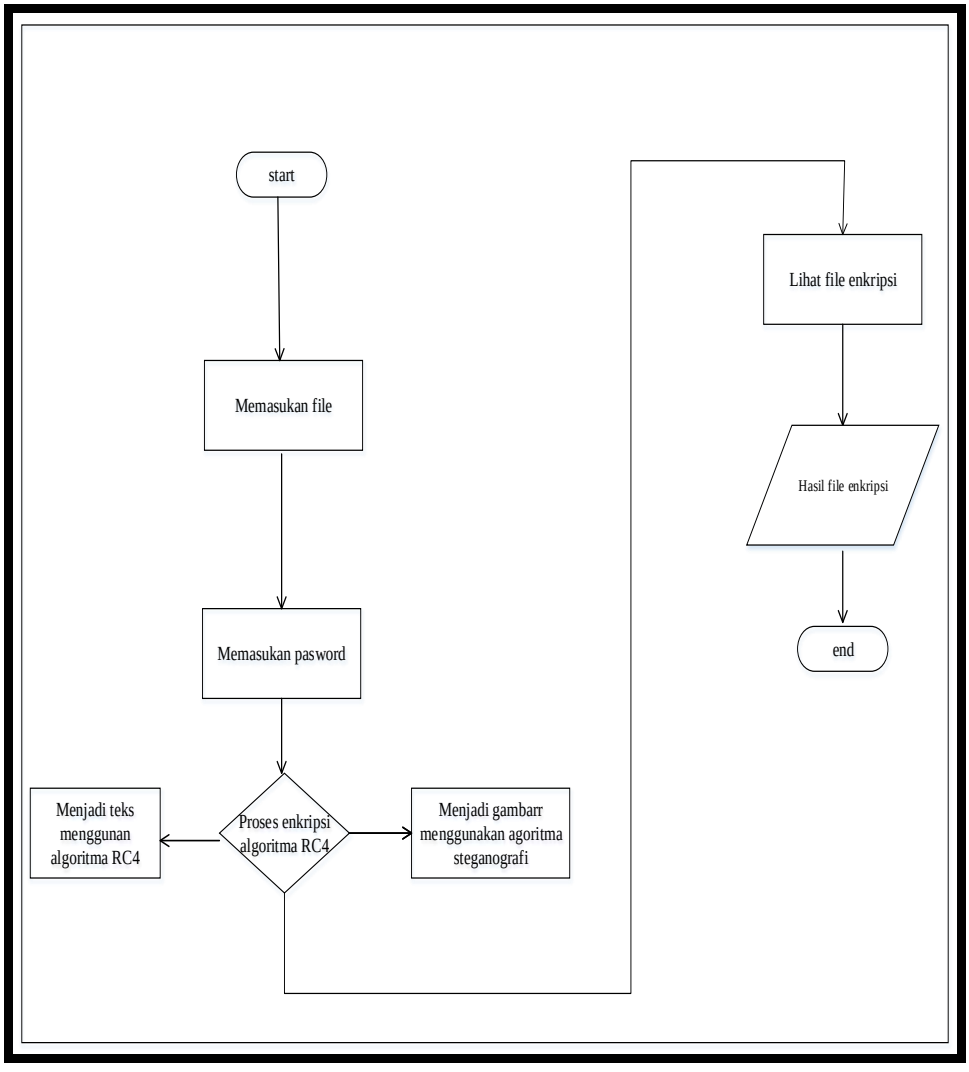
Kompresi adalah sebuah teknik pada ilmu komputer untuk mempercepat proses file. Gambar 5.2 menunjukan skema kompresi dalam algoritma kompresi huffman



Gambar 5.2 Kompresi

b. Enkripsi

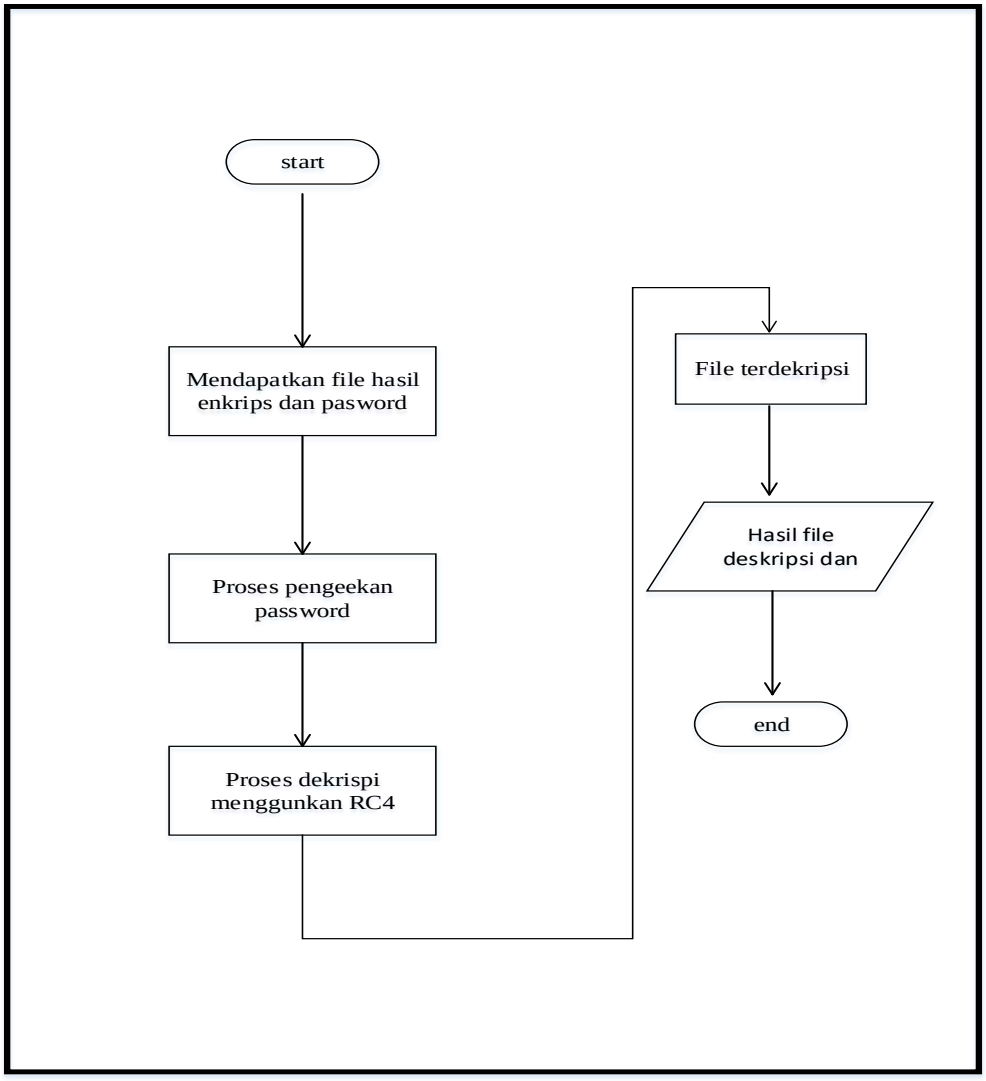
Berikut ini adalah flowchart yang digunakan untuk menelusuri proses program pada aplikasi untuk keamanan file. Flowchart ini merupakan alur jalannya proses enkripsi sebuah file yang ingin dienkripsi. Flowchart proses enkripsi dapat dilihat seperti gambar 5.3



Gambar 5.3 Enkripsi

c. deskripsi

gambar 5.4 merupakan Flowchart proses deskripsi. Flowchart ini menjelaskan proses pengembalian data dari file enkripsi menjadi file asli



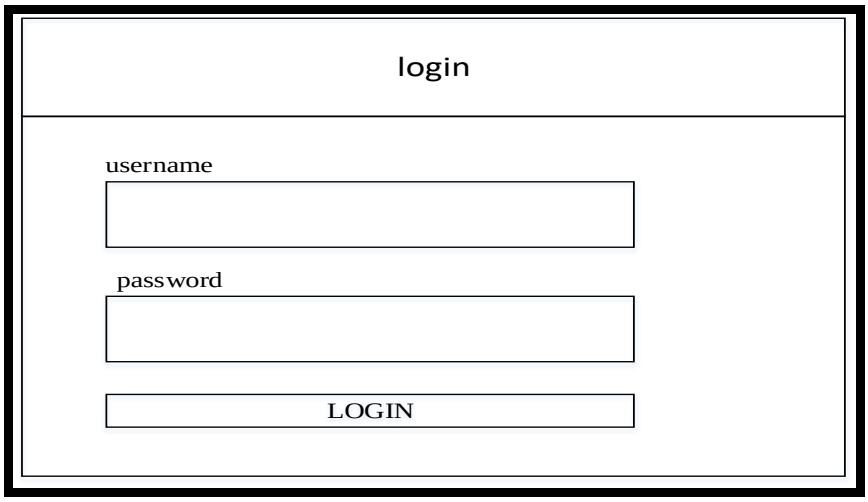
Gambar 5.4 Deskripsi

5.2 Desain *interface*

a. Desain *Form* login

Desain *Form* login ini untuk *security* akses pengguna pada *website*, serta menghindari tindakan yang tidak berkepentingan mengoperasinya. Jika *username* dan *password* yang dimasukan salah, maka kembali ke

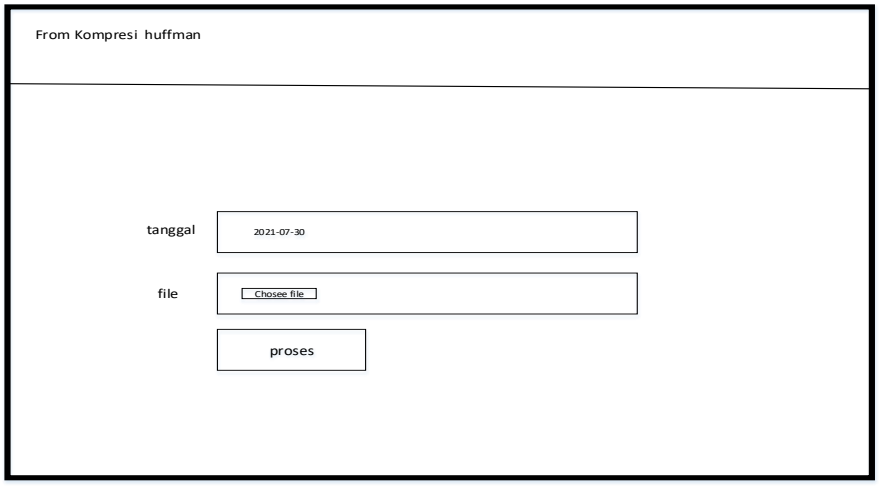
Form login lagi. Adapun desain *Form* login dapat dilihat pada gambar 5.5



Gambar 5.5 Desain *Form* Login

b. Desain *Form* kompresi

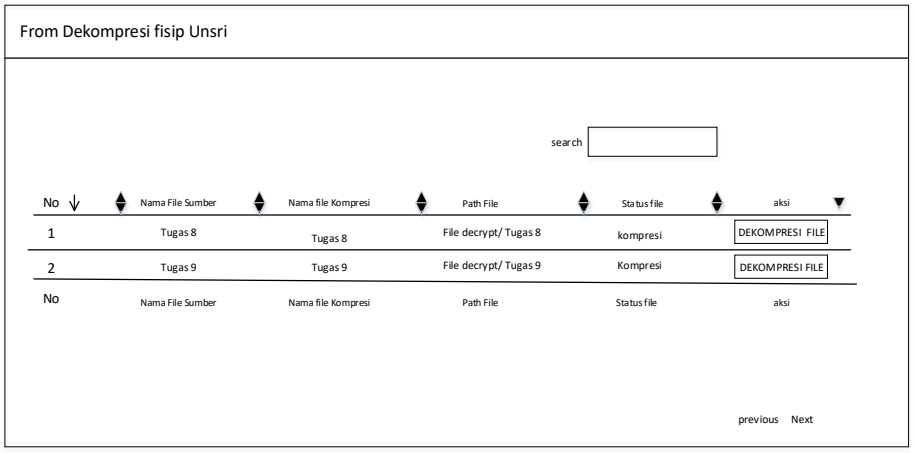
Form kompresi digunakan untuk mempercepat proses suatu data, adapun desain *Form* kompresi dapat dilihat pada gambar



Gambar 5.6 Desain *Form* Kompresi

c. Desain form dekompresi

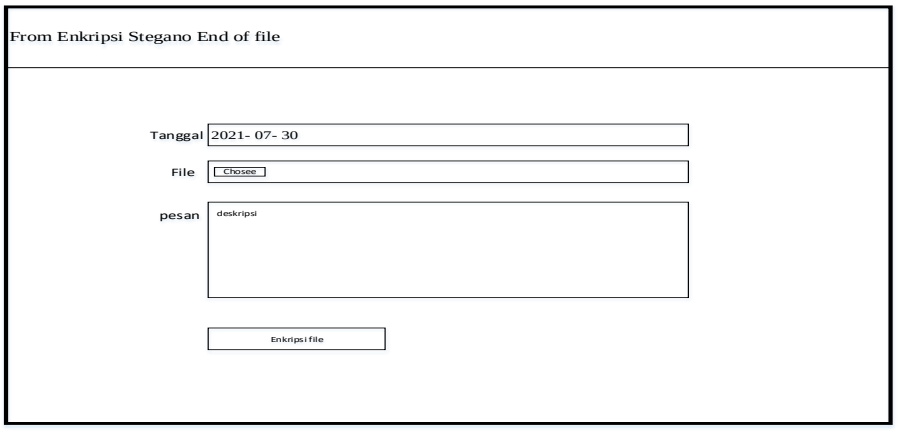
Form dekompresi digunakan untuk mengembalikan data yang telah dikompresi, adapun desain *Form* dekompresi dapat dilihat pada gambar



Gambar 5. 7 Desain *Form* Dekompresi

d. Desain Enkripsi Stegano End of file

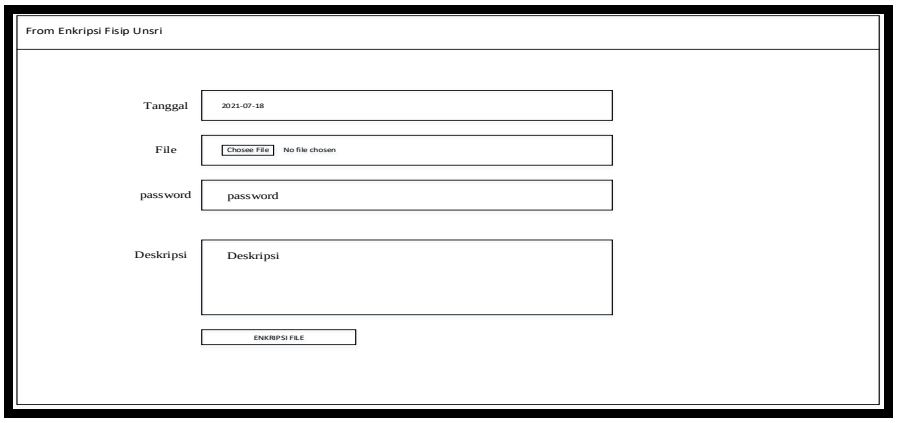
Form enkripsi stegano digunakan untuk menyisipkan suatu pesan kedalam gambar agar pesan tersebut tidak bisa dibaca, adapun desain *Form* enkripsi Stegano End of file dapat dilihat pada gambar



Gambar 5.8 Enkripsi Stegano EOF

e. Desain *Form* Enkripsi

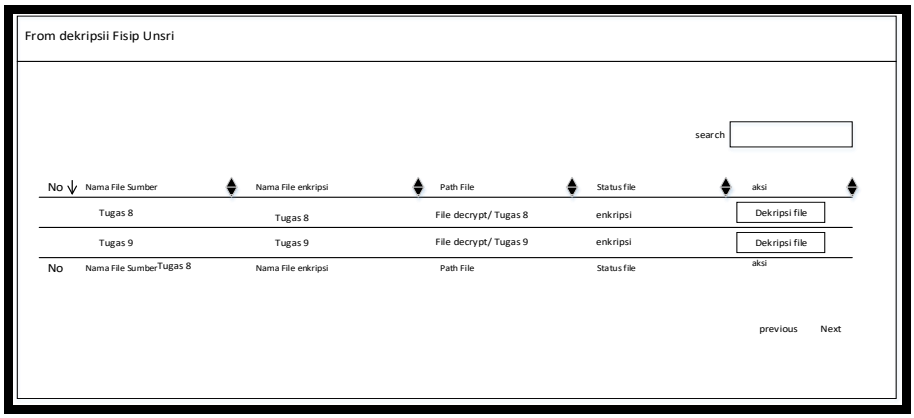
Form enkripsi digunakan untuk mengamankan suatu data dengan membuat data tersebut tidak bisa dibaca, adapun desain *Form* enkripsi dapat dilihat pada gambar 5.9



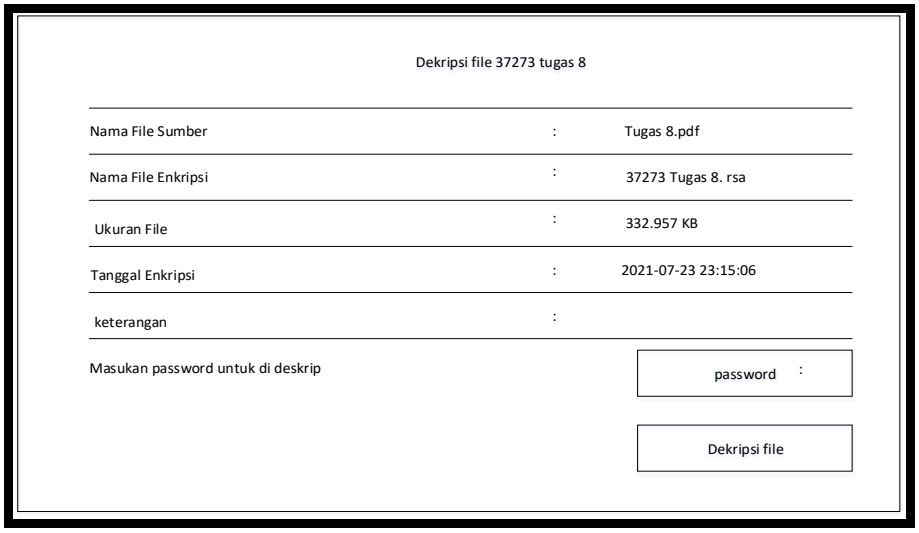
Gambar 5.9 Desain *Form* Enkripsi

f. Desain from dekripsi

Form dekripsi digunakan untuk proses pengembalian data dari file enkripsi menjadi file asli, adapun desain *Form* dekripsi dapat dilihat pada gambar 5.10



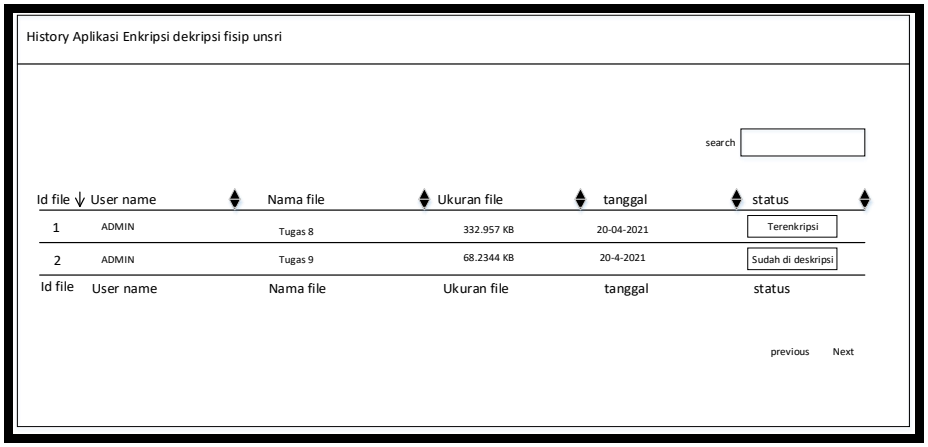
Gambar 5. 10 Desain *Form* Dekripsi



Gambar 5. 11 Desain *Form* Hasil Dekripsi

g. Desain daftar list

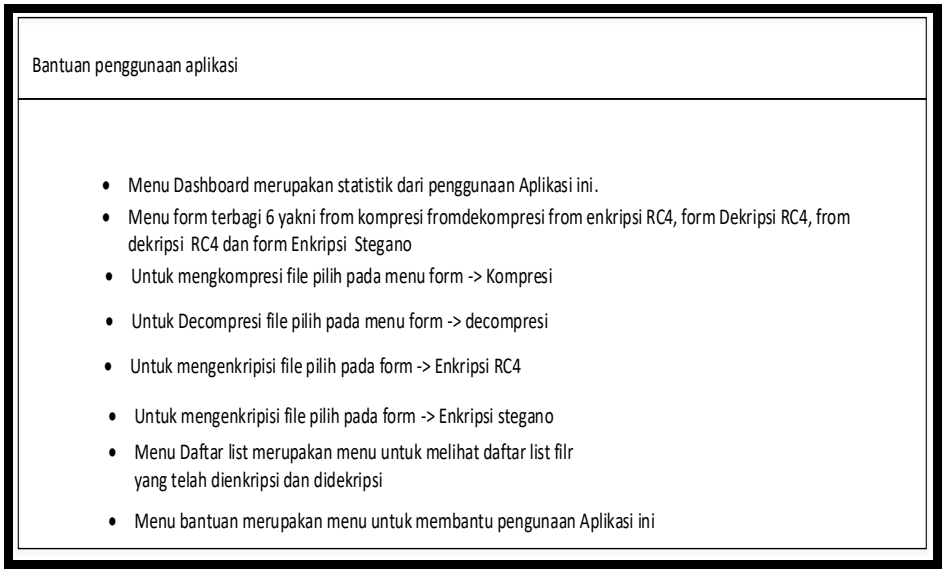
Form daftar list digunakan untuk melihat file yang sudah di enkripsi atau deskripsi , adapun desain *Form* daftar list dapat dilihat pada gambar 5.12



Gambar 5. 12 Desain *Form* Daftar list

h. Desain from bantuan

Form bantuan digunakan untuk meminta bantuan jika ada masalah dalam aplikasi, adapun desain *Form* bantuan dapat dilihat pada gambar 5.13



Gambar 5. 13 Desain Form Bantuan

5.3 Implementasi

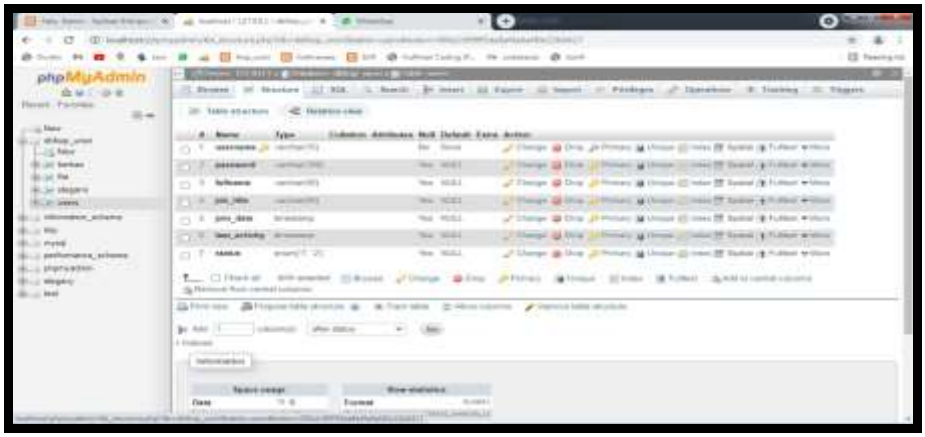
5.3.1 Implementasi Database

Implementasi *database* adalah rancangan *database* yang terdiri *table-table* yang digunakan membangun sebuah aplikasi keamanan data dokumen. *Database* ini akan difungsikan sebagai tempat penyimpanan data. Berikut *table- table* yang tersimpan dalam sebuah *database*:

a) implementasi Tabel data base user

Tabel user berfungsi untuk menyimpan data hak akses yang akan dizinkan melakukan pengolahan data pada aplikasi keamanan data. Tabel

ini akan menyimpan data *username* dan *password* dari masing-masing *user* yang diberikan hak akses untuk masuk kedalam sistem.



Gambar 5.14 Database User

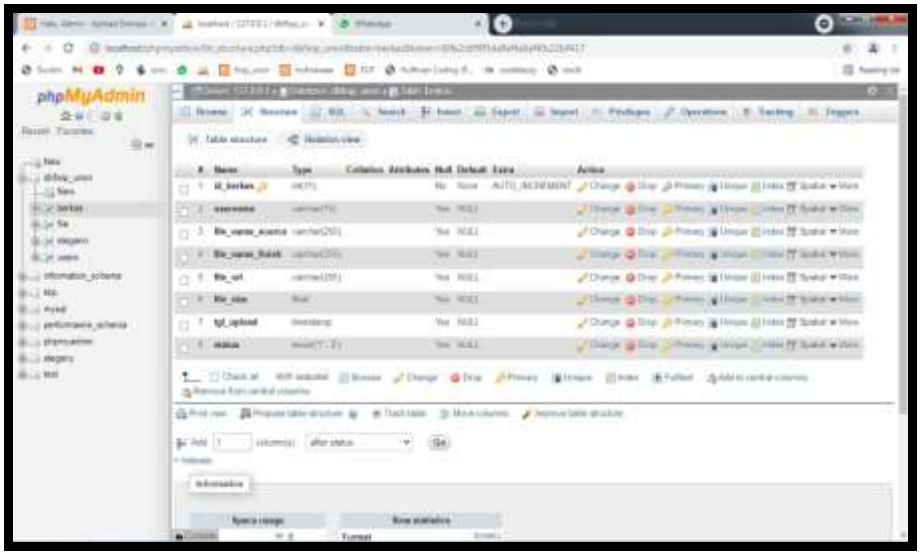
b) Implementasi Tabel Database File

Tabel *file* berfungsi untuk meng enkripsi dan dekripsi algoritma RC4

Gambar 5. 15 Database File

c) Implementasi Tabel Database berkas

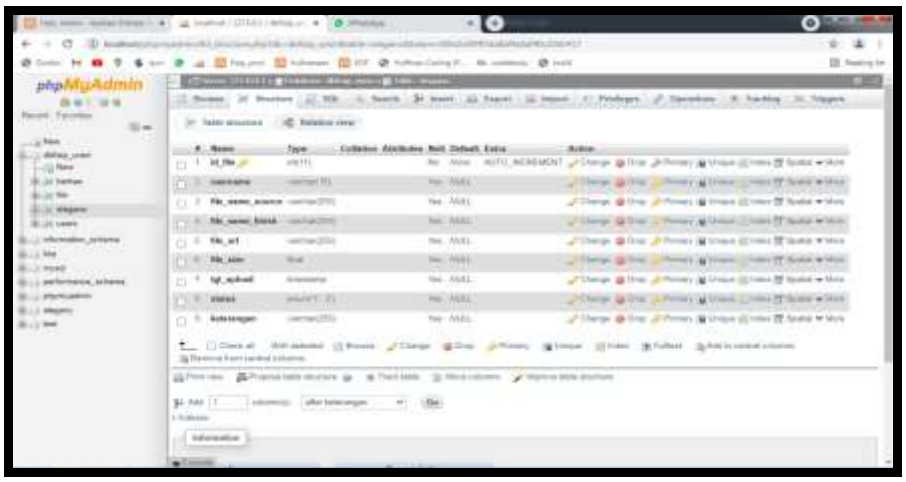
Tabel *berkas* berfungsi untuk menyimpan hasil kompresi dan dekompresi dari algoritma huffman.



Gambar 5. 16 Tabel Database berkas

d) Implementasi Tabel Database stegano

Tabel stegano digunakan untuk penyisipan pesan text kedalam gambar



Gambar 5.17 Tabel Database Stegano

5.3.2 Implementasi *Interface*

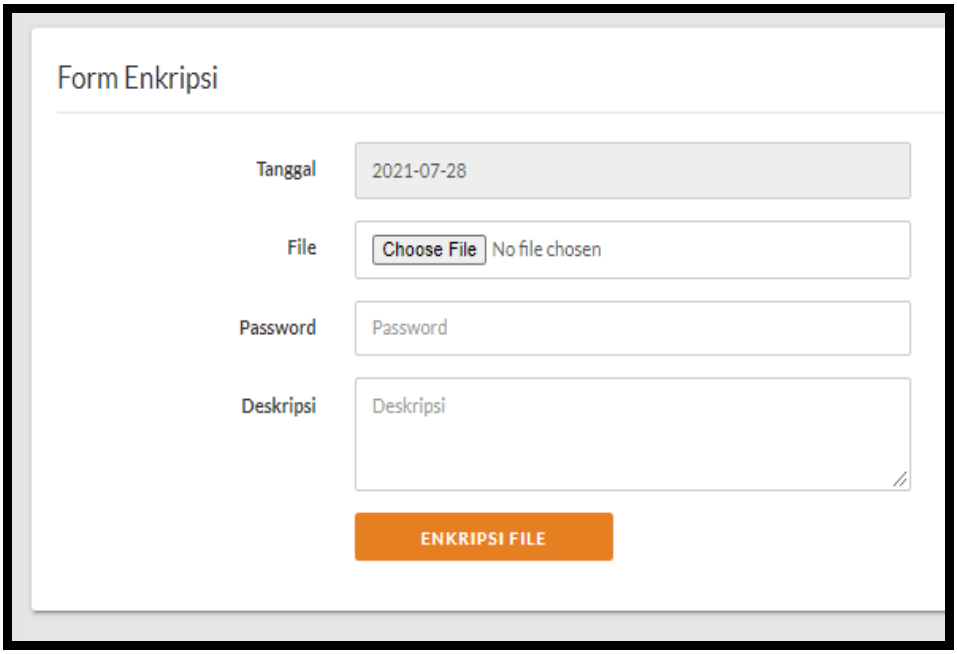
a. Implementasi *Interface* Halaman Login

Halaman *Form* login untuk login ke aplikasi.

Gambar 5. 18 Halaman Login

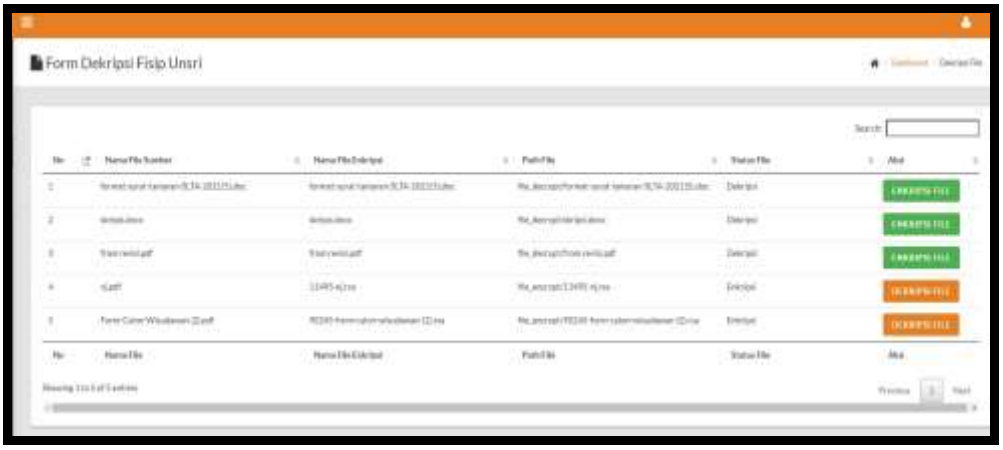
b. Implementasi *Interface* Halaman form enkripsi

Halaman from enkripsi digunakan untuk mengamankan suatu data dengan membuat data tersebut tidak bisa dibaca,



Gambar 5.19 Halaman Enkripsi

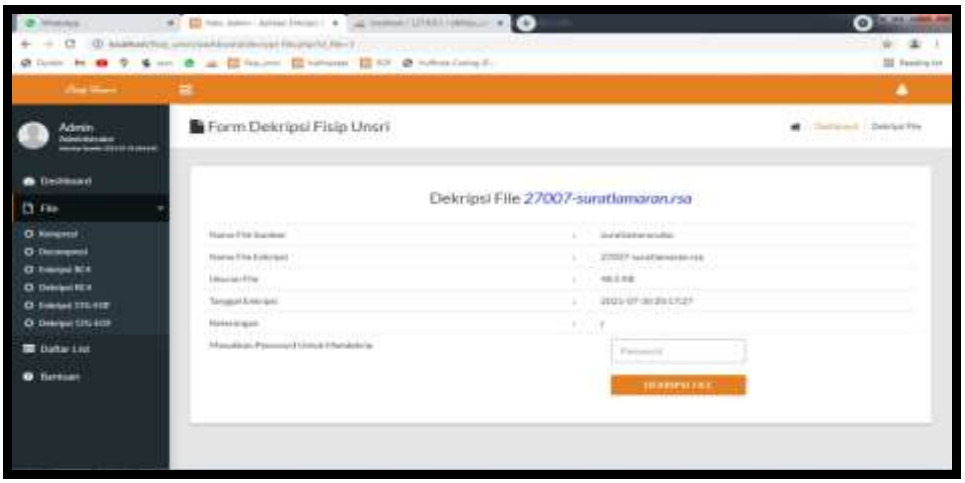
c. Implementasi *interface* halaman hasil Enkripsi & Dekripsi



Gambar 5.20 Halaman Hasil Enkripsi & Dekripsi

d. Implementasi *interface* halaman *form* dekripsi

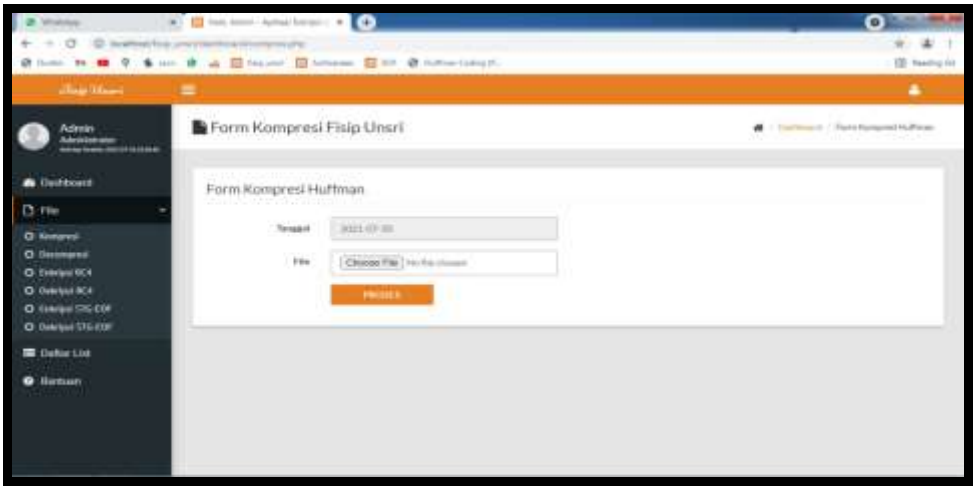
Halaman form dekripsi digunakan untuk mengembalikan file yang sudah di enkripsi menjadi file asli.



Gambar 5. 21 Halaman *Form* Dekripsi

e. Implementasi *interface* halaman *form* kompresi

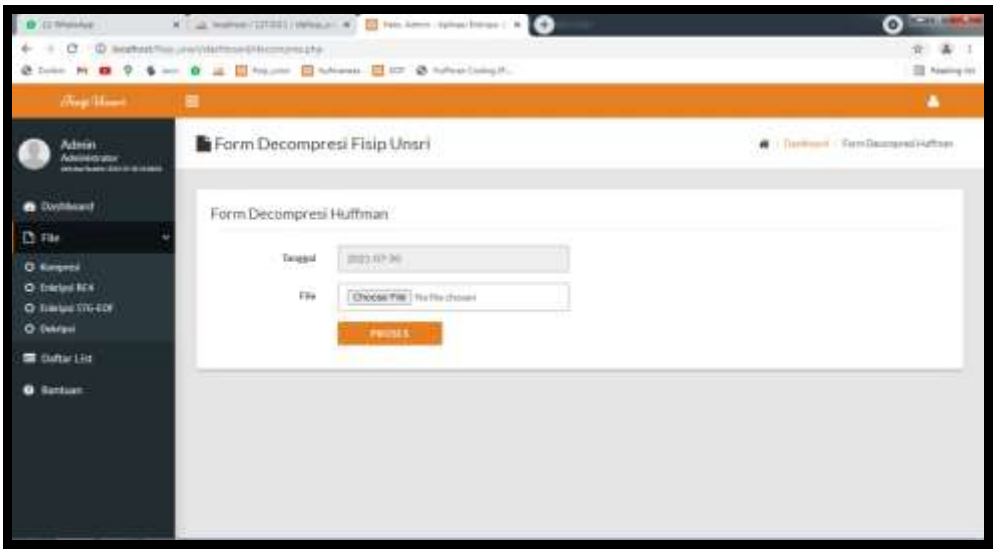
Halaman form kompresi digunakan untuk mempercepat proses file

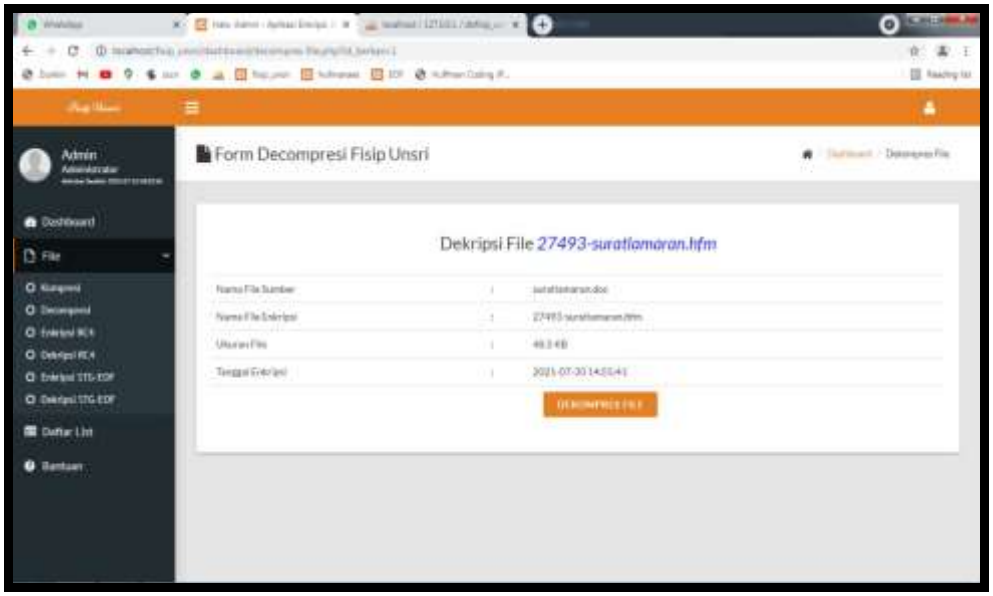


Gambar 5. 22 Halaman *Form* Kompresi

f. Implementasi *interface* halaman *form* dekompresi

Halaman form dekompresi digunakan untuk mengembalikan file yang sudah di kompresi menjadi file asli.

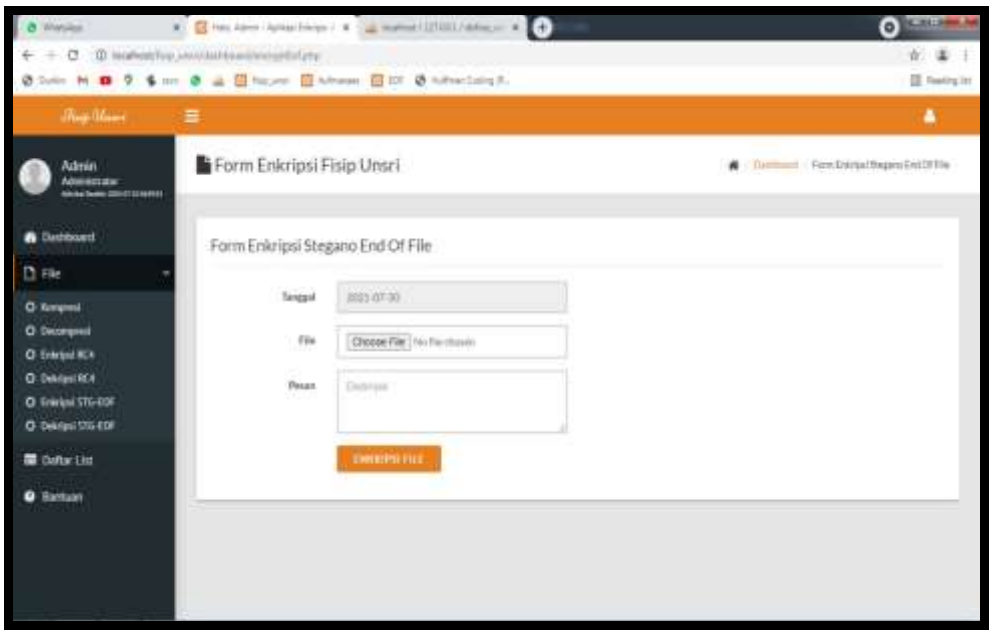




Gambar 5.23 Halaman *Form* Dekompresi

g. Implementasi *interface* halaman *form* enkripsi steganografi

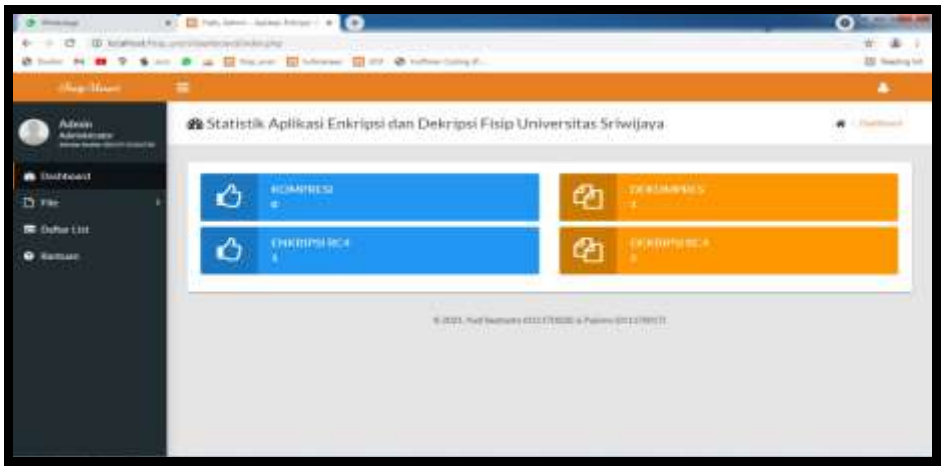
Halaman from enkripsi Steganografi digunakan untuk mengamankan suatu pesan dengan membuat pesan tersebut disisipkan kedalam gambar



Gambar 5.24 Halaman Form Enkripsi Steganografi

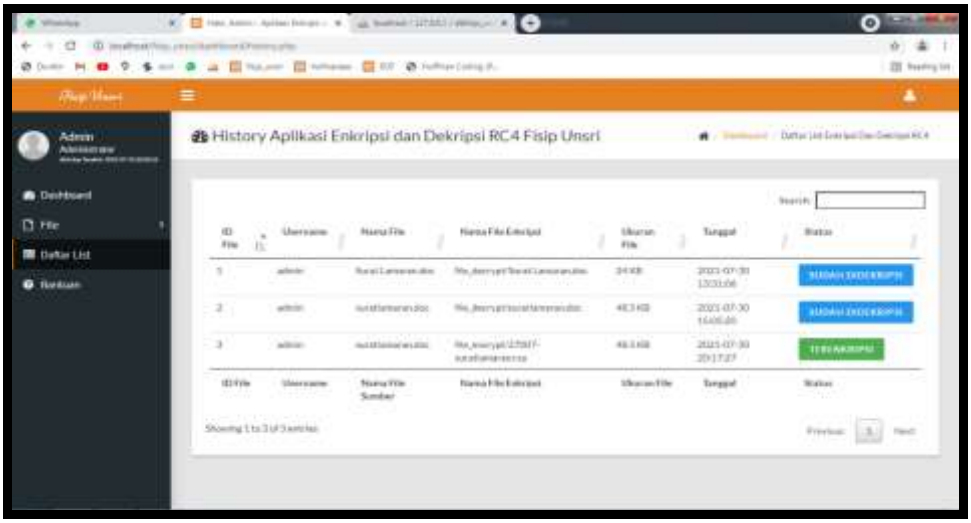
h. Implementasi *interface* halaman *form* menu dasboard

Halaman from menu dasboard digunakan untuk melihat tampilan awal aplikasi yang terdiri dari menu" aplikasi.



Gambar 5. 25 Halaman *Form* Dasboard

j. Implementasi interface halaman daftar list



Gambar 5.26 Halaman Daftar List

Lampiran 1. Data Evaluator

Kode Evaluator	NIDN	Nama	Pendidikan
Ev1	******011170015	Willy	S1: Teknik Infromatika
Ev2	******011170035	Wahyuda	S1: Teknik Infromatika
Ev3	******011170034	Adi Sanjaya	S1: Teknik Infromatika
Ev4	******01119008	Zaki	S1: Teknik Infromatika

Lampiran 2. Hasil Penilaian Evaluator

Kode Evaluator : Ev 1

Nama Evaluator : Welly

NO	Item Pengujian	Keterangan	*Severity Rating* (0-4)
1	Visibility of system status	Antarmuka pada sistem memberikan informasi pada pengguna tentang kondisi suatu proses dalam jangka waktu tertentu.	
Hasil Evaluasi Pada saat pindah menu tidak ada progress bar atau yang menandakan adanya proses yang sedang terjadi sehingga pengguna kebingungan			**3**
Rekomendasi Agar diberikan suatu icon proses sebagai pertanda bahwa aplikasi sedang menjalankan proses			
2	Match between system and the real world	Sistem menggunakan Bahas penggguna dengan kata dan frase yang akrab pada pengguna.	
Hasil Evaluasi Bahasa yang digunakan sudah cukup mudah dipahami akan tetapi perhatikan penggunaan huruf kapital.			**1**
Rekomendasi Bahasa yang bisa disesuaikan yaitu search diganti cari			

Kode Evaluator : Ev 2

Nama Evaluator : Wahyuda

NO	Item Pengujian	Keterangan	*Severity Rating* (0-4)
1	User Control And freedom	Pengguna memiliki kebebasan untuk Kondisi tertentu	
Hasil Evaluasi Tidak adanya fitur untuk mengubah foto profile			**2**
Rekomendasi Agar diberikan suatu setting profile			
2	Aesthetic and minimalist design	Tampilan memiliki estetika/ keindahan	
Hasil Evaluasi Pewarnaan icon pada menu aplikasi kurang begitu menarik			**1**
Rekomendasi Disesuaikan dengan user interface unsri			

Kode Evaluator : Ev 3

Nama Evaluator : Adi Sanjaya

NO	Item Pengujian	Keterangan	*Severity Rating* (0-4)
1	User friendly	Antarmuka pada sistem memberikan informasi pada pengguna tentang kondisi Kenyamanan dan kemudahan.	
Hasil Evaluasi Pada saat klik tombol button disesuaikan dengan icon			**4**
Rekomendasi Agar diberikan suatu icon tombol yang sesuai			
2	Match between system and the real world	Sistem menggunakan Bahasa pengguna, dengan kata dan frase yang akrab pada pengguna.	
Hasil Evaluasi Bahasa yang digunakan sudah cukup mudah dipahami akan tetapi perhatikan penggunaan huruf kapital.			**1**
Rekomendasi Bahasa yang bisa disesuaikan yaitu english dan indonesia			

Kode Evaluator : Ev 4

Nama Evaluator : Zaki

NO	Item Pengujian	Keterangan	*Severity Rating* (0-4)
1	Help users system status	Antarmuka pada sistem memberikan informasi pada pengguna tentang kondisi suatu proses dalam jangka waktu tertentu.	
Hasil Evaluasi Pada saat pindah menu tidak ada progress bar atau yang menandakan adanya proses yang sedang terjadi sehingga pengguna kebingungan			**3**
Rekomendasi Agar diberikan suatu icon proses sebagai pertanda bahwa aplikasi sedang menjalankan proses			
2	Error aplikasi	Error aplikasi telah diuji ,	
Hasil Evaluasi Tidak ditemukan error pada pengujian			**2**
Rekomendasi Peningkatan kecepatan aplikasi			

Lampiran 3. Pengujian Sistem

Pengujian sistem dilakukan untuk memastikan bahwa sistem yang telah dibangun dapat berjalan dengan baik sesuai dengan fungsi-fungsi yang sebelumnya ditentukan pada tahap analisis dan perancangan sistem.

Lampiran 4. Pengujian Hasil Enskripsi

Lampiran 4.1 Pengujian Hasil Enskripsi Sistem

Untuk melakukan proses enkripsi, maka tahap awal yang dilakukan adalah memilih menu enkripsi. Setelah itu lakukan langkah-langkah berikut ini untuk melakukan proses enkripsi.

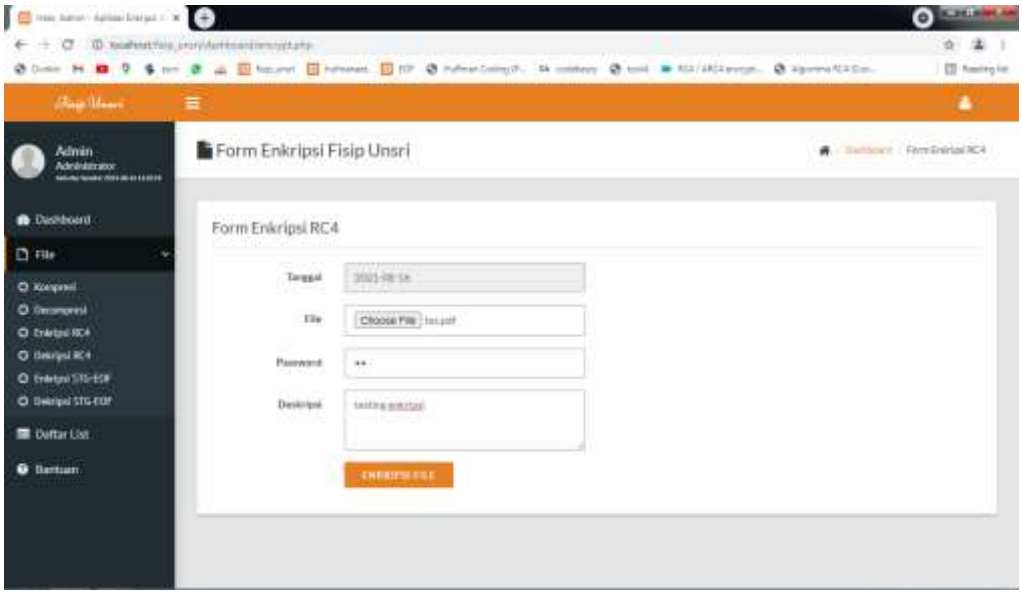
1. Tekan tombol Pilih File untuk memilih file pdf (*.pdf, *.txt dan *.doc) yang akan dienkripsi.

2. Masukkan kunci RC4A pada kolom yang telah disediakan.

3. Masukkan deskripsi file di kolom yang telah disediakan.

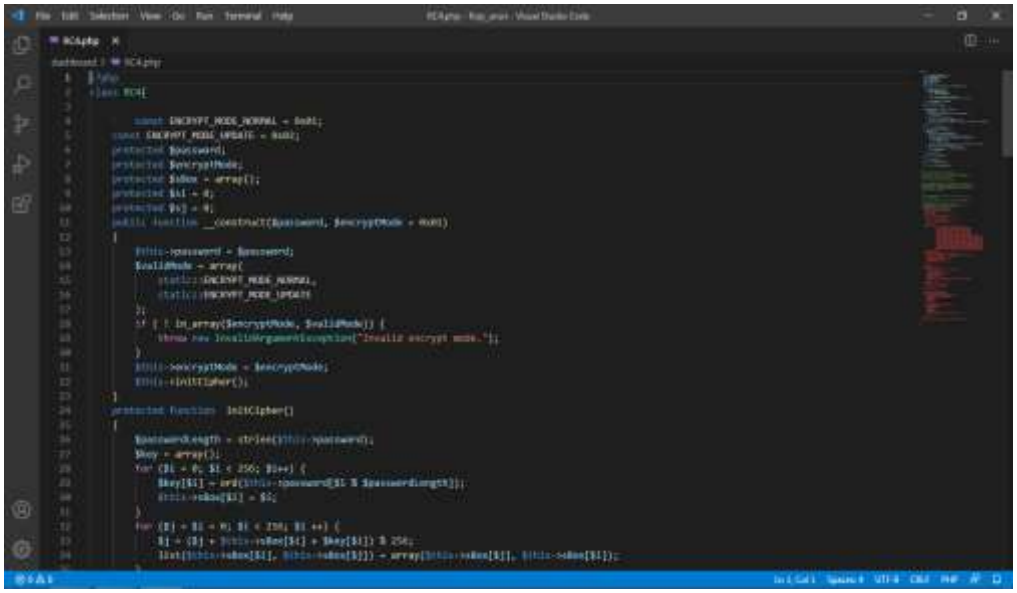
5. Tekan tombol Enkripsi File untuk menyimpan hasil enkripsi.

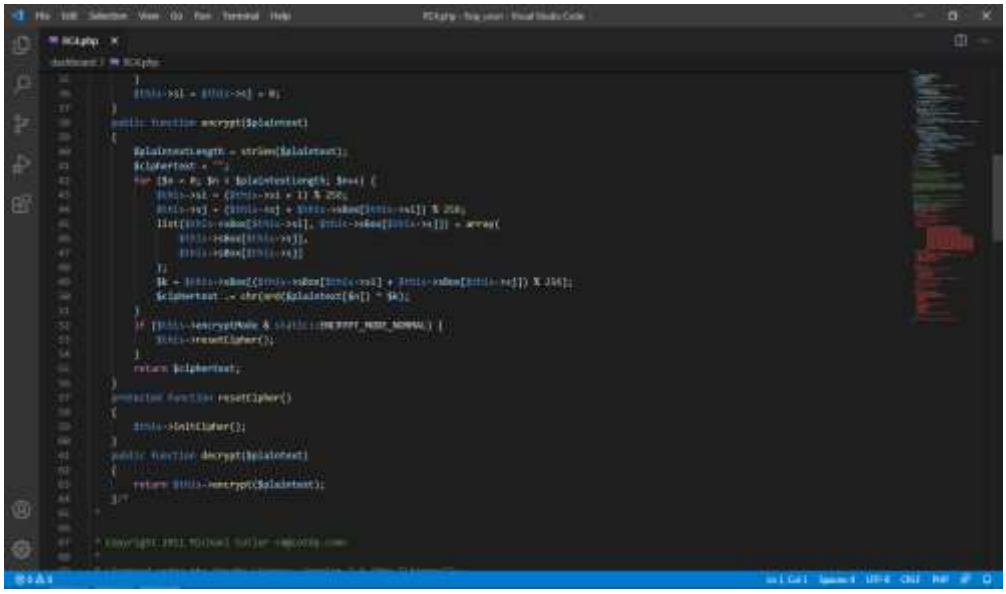
Contoh hasil proses enkripsi dapat dilihat pada Gambar 4.5 dengan masukan file E:\tes.pdf dan kunci “hi”.



Gambar 5.27 Hasil Proses Enkripsi

Gambar 5.28 File Hasil Enkripsi





Gambar 5.29 Gambar Code RC4

Lampiran 5. Pengujian Hasil Enskripsi Manual seperti file PDF.

Berikut adalah contoh hasil enskripsi manual algoritma RC4 dengan mode 4 *byet* (untuk lebih menyederhanakan). Plaintext yang digunakan adalah "HI". Proses KSA dengan masukan K dan S_1 adalah sebagai berikut:

1. Inisialisasi array S1, S1 = {0, 1, 2, 3}
2. Kunci yang digunakan adalah 1 dan 7, *l* = 2 *byte*. Ulangi kunci hingga memeuhi *array* K, K = {1, 7, 1, 7}
3. j = 0

   Iterasi 1 (i = 0, j = 0, S_1 = {0, 1, 2,3})

   j = (j + S_1[i] + K[i mod *l*]) mod 4

   j = (0 + S_1[0] + K[0 mod 2]) mod

   4 j = (0 + 0 + 1) mod 4 = 1

swap $S_1[i]$ dengan $S_1[j]$

swap $S_1[0]$ dengan $S_1[1]$, $S_1 = \{1, 0, 2, 3\}$

4. *Iterasi* 2 (i = 1, j = 1, $S_1 = \{1, 0, 2,3\}$)

   j = 0

   swap $S_1[1]$ dengan

   $S_1[0]$, S = {0, 1, 2, 3}

5. *Iterasi* 3 (i = 2, j = 0, $S_1 = \{0, 1, 2,3\}$)

   j = 3

   swap $S_1[2]$ dengan $S_1[3]$, $S_1 = \{0, 1, 3, 2\}$

6. *Iterasi* 4 (i = 3, j = 3, $S_1 = \{0, 1, 3,2\}$)

   j = 0

   swap $S_1[3]$ dengan $S_1[0]$, $S_1 = \{2, 1, 3, 0\}$

Proses PRGA untuk medapatkan nilai WK adalah sebagai berikut:

1. For WK0 to l – 1, dimana l = 2

   (i = j = 0, $S_1 = \{2, 1, 3, 0\}$)

   Iterasi 1 (WK = 0)

   i = (i + 1) mod 4

   i = (0+1) mod 4 = 1

   j = (j + $S_1[i]$) mod 4 j = (0 + $S_1[1]$) mod 4

   j = (0 + 1) mod 4 = 1

swap $S_1[i]$ dengan $S_1[j]$

swap $S_1[1]$ dengan $S_1[1]$, $S_1 = \{2, 1, 3, 0\}$

output = $S[(S_1[i] + S_1[j]) \bmod 4]$

output = $S[(1 + 1) \bmod 4]$

output = $S[2] = 3$

$WK[0] = 3$

2. *Iterasi* 2 (WK = 1)
   (i = 1, j = 1, $S_1 = \{2, 1, 3, 0\}$)
   $i = (i + 1) \bmod 4$
   $i = (1+1) \bmod 4 = 2$
   $j = (j + S_1[i]) \bmod 4$ j
   $= (1 + S_1[2]) \bmod 4$ j
   $= (1 + 3) \bmod 4 = 0$
   swap $S_1[i]$ dengan $S_1[j]$

   swap $S_1[2]$ dengan $S_1[0]$, $S_1 = \{3, 1, 2, 0\}$

   output = $S[(S_1[i] + S_1[j]) \bmod 4]$

   output = $S[(2 + 3) \bmod 4]$

   output = $S[1] = 1$

   $WK[1] = 1$

   Maka $WK = \{3, 1\}$

Proses KSA dengan Masukan WK dan S_2. Proses ini sama halnya dengan proses sebelumnya, hanya saja jika sebelumnya menggunakan K dan S_1 maka Kali ini yang digunakan adalah WK dan S_2. Pada akhir dari proses ini diperoleh $S2 = \{3, 1, 2, 0\}$. Setelah S_1 dan S_2 diperoleh dari proses KSA, selanjutnya masuk ke tahap PRGA. Proses PRGA dengan Masukan S_1 dan S_2 adalah sebagai berikut:

1. *Iterasi* 1 ($i = j_1 = j_2 = 0$, $S_1 = \{3, 1, 2, 0\}$,

   $S_2 = \{3, 1, 2, 0\}$) $i = (i + 1) \bmod 4$

   $i = (0+1) \bmod 4 = 1$
   $j_1 = (j_1 + S_1[i]) \bmod 4$

   $j_1 = (0 + S_1[1]) \bmod 4$

   $j_1 = (0 + 1) \bmod 4 = 1$

   swap $S_1[i]$ dengan $S_1[j_1]$

   swap $S_1[1]$ dengan $S_1[1]$, $S_1 = \{3, 1, 2, 0\}$

   output $= S_2[(S_1[i] + S_1[j_1]) \bmod 4]$

   output $= S_2[(1 + 1) \bmod 4]$

   output $= S_2[2] = 2$

   $j_2 = (j_2 + S_2[i]) \bmod 4$

   $j_2 = (0 + S_2[1]) \bmod 4$

   $j_2 = (0 + 1) \bmod 4 = 1$
   swap $S_2[i]$ dengan $S_2[j_2]$

   swap $S_2[1]$ dengan $S_2[1]$, $S_2 = \{3, 1, 2, 0\}$

   output $= S_1[(S_2[i] + S_2[j_2]) \bmod 4]$

   output $= S_1[(1 + 1) \bmod 4]$

   output $= S_1[2] = 2$

Selanjutnya *plaintext* H di XOR dengan 2 kemudian di XOR lagi dengan 2, maka : H XOR 2 = 01001000 XOR 00000010 = 01001010 XOR 00000010 = 01001000

2. *Iterasi* 2 (i = 1, j_1 = 1, j_2 = 1, S_1 = {3, 1, 2, 0}, S_2 =

   {3, 1, 2, 0}) i = (i + 1) mod 4

   i = (1+1) mod 4 = 2

   $j_1 = (j_1 + S_1[i]) \bmod 4$
   $j_1 = (1 + S_1[2]) \bmod 4$

   $j_1 = (1 + 2) \bmod 4 = 3$

   swap $S_1[i]$ dengan $S_1[j_1]$

   swap $S_1[2]$ dengan $S_1[3]$, S_1 = {3, 1, 0, 2}

   output = $S_2[(S_1[i] + S_1[j_1]) \bmod 4]$

   output = $S_2[(0 + 2) \bmod 4]$

   output = $S_2[2] = 2$

   $j_2 = (j_2 + S_2[i]) \bmod 4$

   $j_2 = (1 + S_2[2]) \bmod 4$

   $j_2 = (1 + 2) \bmod 4 = 3$
   swap $S_2[i]$ dengan $S_2[j_2]$

   swap $S_2[2]$ dengan $S_2[3]$, S_2 = {3, 1, 0, 2}

   output = $S_1[(S_2[i] + S_2[j_2]) \bmod 4]$

   output = $S_1[(0 + 2) \bmod 4]$

   output = $S_1[2] = 0$

Selanjutnya *plaintext* I di XOR dengan 2 kemudian di XOR lagi dengan 0, maka :I XOR 2 = 01001001 XOR 00000010 = 01001011 XOR 00000000 = 01001011

Tabel 5.1 Proses XOR Keystream dengan Plaintext pada Enkripsi

	H	I
Plaintext	01001000	01001001
Keystream 1	00000010	00000010
	01001010	01001011
Keystream 2	00000010	00000000
Ciphertext	01001000	01001011
	H	K

Sebagaimana terlihat pada Tabel 4.1, plaintext “HI” di xor dengan keystream 1 dan keystream 2 yang telah dihasilkan pada akhir proses PRGA sehingga didapatkan ciphertext pesan berupa “HK”.

Lampiran 6. Pengujian Hasil Dekripsi Manual

Ciphertext yang akan didekripsi adalah *ciphertext* yang telah diperoleh pada enkripsi sebelumnya yaitu “HK”. Proses dekripsi sama seperti proses enkripsi sebelumnya. Jika pada proses enkripsi, *plaintext* di xor dengan *keystream* untuk menghasilkan *ciphertext* sementara pada proses dekripsi, *ciphertext* di xor dengan *keystream* untuk menghasilkan *plaintext*.

Tabel 5. 2 Proses XOR Keystream dengan Ciphertext pada Dekripsi

	H	K
Ciphertext	01001000	01001011
Keystream 1	00000010	00000010
	01001010	01001001
Keystream 2	00000010	00000000
Plaintext	01001000	01001001
	H	I

Sebagaimana terlihat pada Tabel 5.2, *ciphertext* "HK" di xor dengan *keystream* 1 kemudian di xor lagi dengan *keystream* 2 sehingga dihasilkan *plaintext* semula yaitu "HI". *Keystream* 1 dan *keystream* 2 diperoleh dengan cara yang sama seperti pada proses enkripsi sebelumnya.

Lampiran 7. Pengujian Hasil Kompresi

Pengujian kinerja aplikasi dalam melakukan kompresi dan dekompresi, dilihat dari sisi performa yang diukur dengan melihat waktu yang dibutuhkan untuk melakukan kompresi berkas sebelum diunggah dan melakukan dekompresi berkas sesudah diunggah.

Selain mengukur performa aplikasi dari waktu pemrosesan berkas, dilihat pula perbedaan ukuran antara berkas sebelum kompres dengan berkas setelah dikompres.

Tabel 5. 3 Tabel Berkas Yang Digunakan Untuk Pengujian

No	Nama Berkas	Tipe	Ukuran
1	Tes 1	*Docx*	102 KB
2	Tes 2	*Pdf*	233 KB
3	Penguins	*Jpg*	759 KB

Pada subbab ini dipaparkan hasil uji coba dari pengujian yang dilakukan. Hasil pengujian performa kompresi dapat dilihat pada Tabel 5.4

Tipe Berkas	Ukuran Sebelum	Ukuran Sesudah	Rasio Kompresi	Waktu Kompresi	Waktu Total
Docx	102 KB	101 KB	6,6%	0.095	0.118
Pdf	233 KB	232 KB	6,0%	0.020	0.049
Jpg	759 KB	759 KB	0%	0	0.056

Tabel 5.4 Pengujian Performa Kompresi Dalam Detik

Pada hasil kompresi berkas-berkas terlihat bahwa *file* teks biasa atau teks tidak terformat (.jpg) memiliki rasio kompresi yang lebih besar namun membutuhkan waktu kompresi yang lebih lama dibanding *file* teks berformat (.pdf, .docx,) yang memiliki rasio kompresi kecil namun waktu kompresi lebih cepat.

Lampiran 8. Pengujian Hasil Steganografi End Of File (EOF)

Pengujian steganografi digunakan untuk melihat keberhasilan aplikasi dalam melakukan penyisipan pesan pada cover image. Penyisipan pesan teks dilakukan pada cover image dengan memperhatikan ukuran file suatu gambar, dimana pesan yang akan disisipkan diubah menjadi bilangan decimal dan kemudian ditambahkan dengan file gambar dibagian akhir Skenario Pengujian dilakukan dimana hanya mengukur dari segi kualitas dan mutu yang tertuang dalam nilai, pengujiannya sebagai berikut :

Gambar	Ukuran (kb, bytes)	Karakter
	206 KB	10
	195 KB	20
	148 KB	30
	77 KB	40
	204 KB	50
	248 KB	60
	247KB	70

Tabel 5.5 Hasil Pengujian Steganografi

Pengujian terhadap 7 gambar yang sama dengan pesan yang disisipkan berbeda 10 ± 100 karakter dengan perhitungan 1 karakter bernilai 1 byte jadi pesan yang ditambahkan berukuran 10 – 70 bytes, jika dijadikan kb(kilobyte) dibagi dengan nilai 1024 (nilai baca komputer) menjadi 0,0097kb ± 0,0976kb, sehingga memperoleh hasil yang dapat disimpulkan

bahwa pesan yang disisipkan dengan jumlah sedikit antara 10 ± 70 karakter terlihat ukuran gambar tidak mengalami perubahan besar pada ukuran file gambar.

Lampiran 9 pengujian dengan serangan

Pengujian ini dilakukan dengan serangan yang berarti melakukan sesuatu terhadap citra yang telah disisipkan pesan, dan untuk pengujiannya yakni Pengujian robustness dilakukan untuk melihat apakah gambar hasil steganografi yang telah disisipkan pesan rahasia dapat diungkap kembali menjadi pesan rahasia yang akan disampaikan, jika gambar steganografi tersebut akan dilakukan serangan dengan cara memutarkan maupun gambar mengalami pemotongan atau mengubah sudut gambar. Sehingga dalam tabel 8.1 akan menunjukkan hasil pengujiannya.

Gambar	Ukuran (kb, bytes)	Recovery
	206 KB	Gagal
	195 KB	gagal
	148 KB	Berhasil
	77 KB	berhasil

	204 KB	gagal
	248 KB	gagal
	247KB	gagal

Tabel 5.6 pengujian dengan serangan

Analisis hasil pengujian :

Pengujian dengan melakukan serangan dengan cara robustness yaitu ketahanan sebuah citra digital terhadap serangan baik perpotongan maupun perputaran. Hasil pengujian tersebut menunjukan bahwa pada proses perputaran mengakibatkan ukuran pada file citra yang telah disisipkan pesan rahasia mengalami penambahan ukuran file dan berbanding terbalik dengan proses pemotongan citra digital yang mengalami pengurangan ukuran file itu sendiri. Namun dari kesemua pengujian baik pemutaran dan pemotongan citra ditigal mengakibatkan tidak dapat terungkap kembali pesan rahasia yang telah disisipkan sebelumnya.

BAB VI

PENUTUP

6.1 Kesimpulan

Berdasarkan hasil pengujian dan analisa yang telah dilakukan maka dapat diambil beberapa kesimpulan yang diantaranya ialah:

1. Keamanan data pada aplikasi yang menggunakan teknik enkripsi akan dapat lebih terjaga karena RC4 melakukan pengacakan kunci yang sangat rumit sehingga sulit tertembus
2. Aplikasi ini telah memenuhi komponen kriptografi yaitu kerahasiaan, keutuhan dan keaslian data
3. Implementasi RC4 baik itu enkripsi dan deskrip tidak berpengaruh pada perubahan size ukuran data file yang dideskrip dan enkrip

6.2 Saran

Aplikasi ini diharapkan dapat ditingkatkan kinerja sehingga tidak hanya dapat mengenkripsi file dokumen doc, docx, xls, pdf, dan jpg/jpeg saja namun bisa juga untuk file vidio, audio dll.

DAFTAR PUSTAKA

Abdul, Riad, dan Hendra. 2012. *Implementwasi Keamanan SMS Dengan Algoritma RSA Pada Smartphone Android.* Jurnal Ilmiah Fifo Vol. 9, No. 2.

Asrianda. 2017. *Kompresi File Menggunakan Algoritma Huffman Kanonik*. Jurnal PenelitianTeknik Informatika. VOL. 6 No. 2.

Bariah, S. H., & Putera, M. I. 2021. *Penerapan Metode Waterfall Pada Perancangan Sistem Informasi Pengolahan Data Nilai Siswa Sekolah Dasar*. Jurnal Petik Vol. 01 N0.12.

Barovi, Guntoro., & Admojo, F. T. 2020. *Steganographic techniques using modified least significant bit and modification reshape transposition methods*. ILKOM Jurnal Ilmiah. Vol. 13 No. 1.

Dzuljalali, Abdul. 2019. *Rancang Bangun Repository Guru Menggunakan Metode Enkripsi Advance Encryption Standard dan Kompresi Huffman. (Studi Kasus: SMP Muhammadiyah 22 Tanggerang Selatan).* Jurnal Teknik Informatika. Vol 8 No. 12.

Kusniyati, H., Diansyah, S., & Yusuf, R. 2018. *Penerapan Algoritma Rivert Code 4 RC4) {ada Aplikasi Kriptografi Dokumen*. Jurnal PETIR Vol. 11, Issue 1.

Nurhadian dan Pudoli 2016. *Implementasi Keamanan File dengan Kompresi Huffman dan Kriptografi menggunakan Algoritma RC4 serta Steganografi menggunakan End of File Berbasis Desktop pada SMK Negeri 3 Kota.* Jurnal TICOM, Vol. 5 No. 1.

Nurhayati. 2016. *Implementasi Algoritma RC4A Dan MD5 Untuk Menjamin Confidentiality dan Integrity Pada File Teks*. Jurnal Informatika Vol 13 No.1.

Satyapratama, A., Widjayanto., dan Yunus, M. 2016. *Analisis Perbandingan Algoritma LZW dan Huffman Pada Kompresi File Gambar BMP dan PNG*. Jurnal Teknologi Informasi Vol 6 No. 2.