ANALISA PERBANDINGAN STEGANOGRAFI BERKAS DOKUMEN DENGAN METODE END OF FILE DAN LEAST SIGNIFICANT BIT

SKRIPSI

Diajukan Untuk Memenuhi Salah Satu Syarat

Memperoleh Gelar Sarjana Teknik Pada Fakultas Teknik

Universitas Islam Riau

RIKI ANDRIAN

143510241

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS ISLAM RIAU

PEKANBARU

2021

LEMBAR PERNYATAAN BEBAS PLAGIARISME

Saya yang bertanda tangan dibawah ini :

Nama : Riki Andrian

Tempat/Tanggal Lahir : Pulau Maria, 15 Mei 1996

Alamat : Jln.Lintas Sumatera Utara,Gang Jaksa, Kabupaten Asahan, Kecamatan Teluk Dalam

Adalah mahasiswa Universitas Islam Riau yang terdaftar pada :

Fakultas Jurusan : Teknik

Program Studi : Teknik Informatika

Jenjang Pendidikan : Teknik Informatika Strata-1 (S1)

Dengan ini menyatakan dengan sesungguhnya bahwa skripsi yang saya tulis adalah benar dan asli hasil dari penelitian yang telah saya lakukan dengan judul **"Analisa Perbandingan Steganografi Berkas Dokumen Dengan Metode End Of File Dan Least Significant Bit".**

Apabila dikemudian hari ada yang merasa dirugikan dan atau menuntut karena penelitian ini menggunakan sebagian hasil tulisan atau karya orang lain tanpa mencantumkan nama penulis yang bersangkutan, atau terbukti karya ilmiah ini bukan karya saya sendiri atau plagiat hasil karya orang lain, maka saya bersedia menerima sanksi sesuai dengan peraturan perundangan yang berlaku.

Demikian surat pernyataan ini saya buat dengan sesungguhnya untuk dapat digunakan sebagaimana mestinya.

Pekanbaru, Desember 2021

Yang membuat pernyataan

Riki Andrian

ABSTRAK

Didalam pengamanan suatu berkas dokumen, sangatlah dibutuhkan menggunakan sebuah metode atau algoritma ilmu kriptografi, agar berkas dokumen tersebut bisa terjamin keamanannya, diantara begitu banyaknya metode/algoritma yang dapat dijadikan pengaman berkas dokumen tersebut adalah algoritma Steganografi (*Least Significant Bit*). Algoritma steganografi mempunyai metode yang bisa digunakan seperti LSB (*Least Significant Bit*) dan EOF (*End Of File*). Keunggulan lebih banyak didapatkan pada metode LSB dibandingkan dengan metode EOF dikarenakan citra setelah disisipkan pesan hanya mengalami sedikit penurunan kualitas yang tidak begitu berpengaruh secara signifikan bila dilihat oleh mata manusia, sedangkan mettode EOF mengalami perubahan yang signifikan pada ukuran citra, sehingga untuk metode penyisipan pesan pada gambar lebih baik jika digunakan menggunakan metode LSB.

Kata Kunci : ***enkripsi, steganografi, dekripsi, end of file, least significant bit.***

ABSTRACT

In securing a document file, it is necessary to use a cryptographic method or algorithm, so that the document file can be guaranteed its security, among the many methods/algorithms that can be used as a security for the document file is the Steganography (Least Significant Bit) algorithm. The steganographic algorithm has methods that can be used such as LSB (Least Significant Bit) and EOF (End Of File).More advantages are obtained in the LSB method compared to the EOF method because the image after the message is inserted only experiences a slight decrease in quality which does not have a significant effect when viewed by the human eye, while the EOF method experiences a significant change in the size of the image, so for the message insertion method on the picture is better if used using the LSB method.

***Keyword* : *enkripsi, steganografi, dekripsi, end of file, least significant bit*.**

KATA PENGANTAR

Dengan mengucapkan puji syukur kepada Allah Subhanahu Wa Ta'ala yang telah melimpahkan segala rahmat dan hidayah-Nya kepada penulis, sehingga penulis dapat menyelesaikan tugas akhir yang berjudul "Analisa Perbandingan Steganografi berkas Dokumen Dengan Metode *End Of File* Dan Lsb".

Pada kesempatan ini penulis menyampaikan penghargaan yang setinggi-tingginya kepada semua pihak yang telah memberikan kontribusinya sebelum dan selama pengerjaan tugas akhir ini. Atas semua bantuan, bimbingan, arahan, dukungan dan fasilitas yang telah diberikan, penulis mengucapkan terima kasih.

Pengerjaan tugas akhir ini dilakukan dengan semaksimal mungkin oleh penulis, tetapi penulis menyadari bahwa hasil yang diperoleh masih jauh dari sempurna. Oleh karena itu, saran dan kritik yang membangun sangat penulis harapkan untuk kesempurnaan tugas akhir ini. Besar harapan penulis agar tugas akhir ini dapat bermanfaat bagi pendidikan, khususnya di Fakultas Teknik Universitas Islam Riau.

Akhirnya segala hal yang benar dan terealisasi pada tulisan ini semata-mata karena Allah Subhanahu Wa Ta'ala. Segala kesalahan yang ada semuanya karena kekurangan dan keterbatasan penulis.

Pekanbaru, Desember 2021

Riki Andrian

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Pada umumnya seseorang berharap pesan yang dikirim kepada orang lain tidak dibaca oleh orang yang tidak berhak, terutama pesan yang bersifat rahasia atau pribadi yang hanya boleh diketahui oleh pihak pengirim dan pihak penerima pesan atau kalangan terbatas saja. Untuk menjaga kerahasiaan pesan yang dikirimkan tersebut diperlukan suatu teknik, salah satunya adalah menggunakan steganografi. Steganografi (*steganography*) adalah ilmu dan seni menyembunyikan pesan rahasia (*hiding message*) di dalam pesan lain sehingga keberadaan pesan rahasia tersebut tidak dapat diketahui.

Didalam pengamanan suatu berkas dokumen, sangatlah dibutuhkan menggunakan sebuah metode/algoritma ilmu kriptografi, agar berkas dokumen tersebut bisa terjamin keamanannya, diantara begitu banyaknya metode/algoritma yang dapat dijadikan pengaman berkas dokumen tersebut adalah algoritma Steganografi (*Least Significant Bit*). Algoritma steganografi mempunyai metode yang bisa digunakan seperti LSB (*Least Significant Bit*) dan EOF (*End Of File*).

Kedua algoritma ini mempunyai model yang sangat berbeda dalam proses penyamaran dan penyembunyian data (Gunawan, 2018). Lain dari itu, algoritma ini juga masih digunakan untuk pengembangan didalam ilmu steganografi itu sendiri agar dapat menghasilkan model-model terbaru dari algoritma steganografi. Dalam penelitian sebelumnya dengan judul Pengamanan Acakan BISS Menggunakan Algoritma RSA, telah dijelaskan tentang bagaimana proses

penyisipan teks kedalam sebuah video, lalu mengacaknya, akan tetapi akan terlihat perbedaan yang sangat jauh dari segi ukuran video yang sudah disisipkan oleh teks.

Pada penelitian ini akan membandingkan metode sehingga mampu mengenkripsi dan deskripsi data tanpa mengubah integritas data tersebut. Berdasarkan latar belakang permasalahan di atas, maka penelitian ini akan menggunakan metode EOF (End Of File), atau LSB (Least Significant Bit), Membandingkan dua metode tersebut pada berkas dokumen. Sehingga penulis menarik sebuah judul yaitu "Analisa Perbandingan Metode Steganografi *End Of File dan Least Significant Bit* Pada Berkas Dokumen".

1.2 Identifikasi Masalah

Adapun identifikasi masalah yang dapat diambil dari latar belakang tersebut adalah sebagai berikut :

1. Didalam pengamanan suatu berkas dokumen, sangatlah dibutuhkan menggunakan sebuah metode/algoritma ilmu kriptografi, agar berkas dokumen tersebut bisa terjamin keamanannya.
2. Bagaimana sesorang yang tidak memiliki hak akses, dapat melakukan pembobolan data, terutama data pesan teks. Hal ini dapat memicu kesalah pahaman antara si pengirim dan si penerima.
3. Menguji perbandingan proses enkripsi menggunakan metode steganografi *End Of File* dan *Least Significant* pada berkas dokumen.

1.3 Rumusan Masalah

Pada penelitian ini, penulis merumuskan masalah sebagai berikut:

1. Bagaimana membandingkan proses enkripsi menggunakan Metode Steganografi *End Of File* dan *Least Significant* pada berkas dokumen?
2. Bagaimana merancang aplikasi dengan membandingkan 2 metode steganografi *End Of File* dan *Least Significant Bit* pada berkas dokumen?

1.4 Batasan Masalah

Adapun batasan masalah dari penelitian ini adalah :

1. Wadah yang digunakan untuk menyisipkan data adalah media dalam bentuk dokumen.
2. Hasil *file output* dokumen yang disisipkan pesan langsung disimpan.
3. Pada implementasi perangkat lunak dengan 2 metode steganografi *End Of File* dan *Least Significant Bit* data rahasia di enkripsi terlebih dahulu baru disisipkan pada file dokumen.

1.5 Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah :

1. Untuk mengimplementasikan perbandingan Metode Steganografi *End Of File* dan *Least Significant* Pada Berkas dokumen Untuk merancang aplikasi steganografi yang dapat meningkatkan keamanan dan kerahasiaan data pengguna.
2. Mengetahui kelebihan dan kekurangan Metode Steganografi *End Of File* dan *Least Significant Bit* serta dapat diimplementasikan untuk diterapkan dalam pengunaanya.

1.6 Manfaat penelitian

Adapun manfaat dari penelitian yang dilakukan adalah :

1. Agar dapat mengamankan pesan yang di kirim pengirim ke pada penerima data rahasia sehingga aman saat disampaikan kepada penerima.
2. Agar dapat digunakan sebagai salah satu program oprasional untuk penigkatan efisiensi keamanan penyimpanan maupun pengiriman data khusus nya dokumen teks.
3. Agar dapat mengetahui perbandingan antara 2 metode steganografi yang berbeda kedalam sebuah aplikasi.

BAB II

LANDASAN TEORI

2.1 Tinjauan Pustaka

Studi keperpustakaan pertama adalah berdasarkan penelitian yang dilakukan oleh (Indra Gunawan, 2018), Didalam Dunia Teknologi Informasi ilmu komputer, pengamanan data sesuatu hal yang sangat penting supaya data tidak dapat disalah gunakan beberapa pihak yang belum/bukan memiliki hak. Dalam pengiriman berkas dokumen sangatlah dibutuhkan suatu pengamanan supaya berkas dokumen tersebut bisa diterima oleh yang memiliki hak untuk menerimanya. Oleh karena itu sangatlah dibutuhkan suatu proses penyandian/enkripsi berkas dokumen. Diantar ilmu kriptografi yang dapat mengamankan berkas dokumen tersebut diantaranya Algoritma Steganografi LSB (*Least Significant Bit*). Analisa ini bertujuan untuk meningkatkan pengamanan berkas dokumen dengan cara menyandikan sebuah berkas dokumen tersebut, lalu memberikan sebuah sandi kedalam berkas dokumen tersebut.

Studi ke perpustakaan kedua adalah berdasarkan penelitian yang dilakukan oleh (Akik Hidayat, 2009), Keamanan dan kerahasiaan sangat dibutuhkan dalam dunia komunikasi khususnya dalam dunia komunikasi digital. Diperlukan metode khusus untuk menjamin keamanan informasi, supaya informasi hanya dapat di mengerti oleh pihak yang dituju. Teknik yang umum digunakan adalah dengan mengacak data atau Kriptografi. Tetapi informasi yang diacak sering menimbulkan kecurigaan, maka dibutuhkan teknik lain yaitu dengan penyembunyian data atau Steganografi. Teknik Steganografi dapat diterapkan

sebagai kelanjutan dari Kriptografi, dan kombinasi dari keduanya akan menghasilkan tingkat keamanan data yang sangat tinggi. Akan diperlihatkan menggabungkan teknik Kriptografi dan Steganografi untuk menjaga keamanan data teks sekaligus menyisipkan data teks tersebut dalam gambar digital tanpa mengubah gambar tersebut secara visual,sehingga menghasilkan sebuah metode Steganografi yang optimal untuk menyembunyikan suatu pesan teks di dalam sebuah gambar digital.

Studi ke perpustakaan ketiga adalah berdasarkan penelitian yang dilakukan oleh (Vicky Indriyono, 2016), Meningkatnya perkembangan komunikasi data membuat aspek keamanan dan kerahasiaan informasi menjadi hal yang penting untuk diperhatikan. Karena banyak informasi yang bersifat rahasia yang beresiko diambil oleh pihak yang tidak berkepentingan. Untuk menjaga keamanan dan kerahasiaan data dalam bidang teknologi informasi dapat dilakukan dengan menerapkan teknik kriptografi dan steganografi. Kriptografi adalah ilmu dan seni untuk menjaga kerahasiaan data, sedangkan steganografi adalah ilmu dan seni yang digunakan untuk menyembunyikan pesan ke dalam suatu media. Media tersebut dapat berupa gambar, audio, atau video. Dalam penelitian ini, akan dibuat sebuah aplikasi yang mengkombinasikan algoritma Caesar cipher dan steganografi *End Of File* (EOF). Tahap uji coba dimulai dengan melakukan pengacakan isi file bertipe teks dengan metode Caesar cipher. Setelah isi file teracak, selanjutnya file disisipkan ke dalam gambar yang telah terpilih. Untuk dapat membaca pesan kembali, maka pesan di dekripsi dan hasil dekripsi akan disimpan pada file teks yang baru.

2.1 Dasar Teori

2.1.1 Teori Kriptografi

Kriptografi dapat didefinisikan sebagai seni maupun ilmu yang menghasilkan pesan yang rahasia. Sebuah pesan asli yang disebut sebagai *plaintext* disandikan menjadi pesan yang tersandi yang disebut sebagai *ciphertext* melalui proses enkripsi dan *ciphertext* dipulihkan menjadi *plaintext* kembali melalui proses dekripsi. Kriptografi memiliki beragam algoritma yang telah banyak digunakan sebagai keamanan untuk informasi. Algoritma kriptografi dikelompokkan ke dalam dua jenis yaitu algoritma kriptografi klasik dan algoritma kriptografi modern. Dalam pengoperasiannya, algoritma kriptografi klasik bekerja menggunakan mode karakter sedangkan algoritma kriptografi modern bekerja menggunakan mode *bit*.

Kriptografi memiliki dua konsep utama, yaitu enkripsi (*encryption*) dan dekripsi (*decryption*). Enkripsi adalah proses penyandian *plainteks* menjadi *cipherteks*, sedangkan dekripsi adalah proses mengembalikan *cipherteks* menjadi *plainteks* semula. Enkripsi dan dekripsi membutuhkan kunci sebagai parameter yang digunakan untuk transformasi.

2.1.2 Metode *Least Significant Bit* (LSB)

Salah satu metode steganografi yang banyak digunakan adalah metode modifikasi LSB (*Least Significant Bit*). Metode modifikasi LSB tergolong metode yang menggunakan teknik substitusi. Metode LSB terutama digunakan untuk steganografi berbasis media (media-*based steganography*). Metode LSB

menyembunyikan data rahasia dalam *bit-bit* tak signifikan (*least significant bit*) dari berkas wadah (*cover*). Pengubahan tersebut pada dasarnya memberikan pengaruh terhadap berkas wadah, tetapi karena perubahan yang terjadi sangat kecil, sehingga tidak tertangkap oleh indra manusia. Kenyataan inilah yang akhirnya dimanfaatkan sebagai teknik penyembunyian data atau pesan (steganografi). Sebagai ilustrasi cara penyimpanan data dengan metode LSB.

2.1.3 Metode *End of File* (EOF)

Secara umum teknik steganografi menggunakan *redundant bits* sebagai tempat menyembunyikan pesan pada saat dilakukan kompresi data, dan kemudian menggunakan kelemahan indra manusia yang tidak *sensitive* sehingga pesan tersebut tidak ada perbedaan yang terlihat. 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 file. Perhitungan ukuran file yang telah disisipkan data sama dengan ukuran file sebelum disisipkan data ditambah ukuran data rahasia yang telah diubah menjadi *encoding file*. Dengan metode EOF, secara umum media steganografi (file yang akan disisipi data) memiliki struktur seperti gambar dibawah ini :

Gambar 2.1 Struktur File Steganografi Dengan Metode *End of File*

2.1.4 Teori Steganografi

Steganografi adalah ilmu dan seni untuk menyembunyikan suatu informasi "rahasia" didalam suatu informasi lainnya. Steganografi juga merupakan teknik menyembunyikan data dalam data lain yang akan ditumpanginya tanpa mengubah data tersebut sehingga pada data yang ditumpangi sebelum dan setelah proses penyembunyian hampir sama.

Menurut (Cahyadi, 2012), bahwa ada beberapa contoh media penyisipan pesan rahasia yang digunakan dalam teknik *steganography* antara lain:

1. Teks

Dalam algoritma *steganography* yang menggunakan teks sebagai media penyisipan biasanya digunakan teknik NLP (*Natural Language Processing*) sehingga teks yang telah disisipi pesan rahasia tidak akan mencurigakan untuk orang yang melihatnya.

2. Gambar

Format gambar paling sering digunakan, karena format ini merupakan salah satu format *file* yang sering dipertukarkan dalam dunia internet. Alasan lainnya adalah banyaknya tersedia algoritma *Steganography* untuk media penampung yang berupa citra.

3. Audio

Format ini pun sering dipilih karena biasanya berkas dengan format ini berukuran relatif besar, sehingga dapat menampung pesan rahasia dalam jumlah yang besar pula.

4. Video

Format ini memang merupakan format dengan ukuran *file* yang relatif sangat besar namun jarang digunakan karena ukurannya yang terlalu besar sehingga mengurangi kepraktisannya dan juga kurangnya algoritma yang mendukung format ini.

Sebuah steganografi memiliki tiga aspek yang dapat menentukan berhasil tidaknya sebuah steganografi dalam melakukan pekerjaannya (Emardi dkk, 2004):

1. Kapasitas (*capacity*)

   Kapasitas merujuk pada jumlah informasi yang bisa disembunyikan dalam medium *cover*. Keamanan adalah ketidakmampuan pengamat untuk mendeteksi pesan tersembunyi dan ketahanan dalam jumlah modifikasi medium stego yang bisa bertahan sebelum musuh merusak pesan rahasia tersembunyi tersebut.

2. Keamanan (*security*)

   Keamanan dari sistem steganografi klasik mewujudkan kerahasiaan sistem *encodingnya.*

3. Ketahanan (*robustness*)

   Ketahanan mangacu pada data citra penampung seperti pengubah kontras, penajaman, rotasi, perbesar gambar, pemotongan dan lain-lainnya. Bila pada citra dilakukan operasi pengolahan citra, maka data yang disembunyikan tidak rusak.

2.1.5 *Unified Modelling Language* (UML)

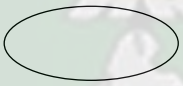
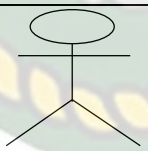
Unified Modeling Language (UML) adalah Bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk menudukung pembangunan sistem (Windu dan Grace dalam Suendri 2018). *Unified Modeling Language* (UML) adalah sebuah Bahasa yang berdasarkan grafik untuk memvisualisasi, menspesifikasikan, membangun, dan pendokumetasian dari sebuah sistem pembangunan *software* berbasis OO (*Object-*

Oriented). UML sendiri juga memberikan standar penulisan sebuah sistem *blue print*, meliputi konsep bisnis proses, penulisan kelas-kelas dalam Bahasa program yang spesifik, skema *database*, dan komponen-komponen yang diperlukan dalam sistem *software* (Siti Fatima dalam Suendri 2018).

2.1.6 *Use Case Diagram*

Use Case Diagram merupakan permodelan untuk kelakuan (*behavor*) sistem informasi yang akan dibuat. *Use Case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar *use case* digunakan untuk menggunakan fungsi-fungsi tersebut (Shalahuddin dalam Umar Al Faruq, 2015). *Use case* terdiri dari beberapa *symbol*, yaitu bisa dilihat pada tabel 2.1 dibawah ini :

Tabel 2.1 Simbol pada *Use Case*

No	Nama	*Symbols*	Keterangan
1	*Use Case*		Abstraksi dari interaksi antara *system* dan *actor*
2	*Actor*		Mewakili peran orang, sistem yang lain atau alat ketika berkomunikasi dengan *use case*
3	*Relationship*	 * *	Penghubung antara objek satu dengan yang lain.

2.1.7 *Class Diagram*

Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metode/fungsi). *Class* diagram adalah sebagai suatu set objek yang memiliki atribut dan perilaku yang sama (Whitten dalam Suendri, 2018). Diagarm Class bersifat statis, menggambarkan hubungan apa yang terjadi bukan apa yang terjadi jika mereka berhubungan. Diagram *class* memiliki tiga area pokok yaitu:

1. Nama, diagram *class* harus memiliki nama.
2. Atribut, adalah kelengkapan yang melekat pada *class*. Nilai dari suatu *class* hanya bisa diproses sebatas atribut yang dimiliki.
3. Operasi, adalah proses yang dapat dilakukan oleh sebuah *class*, baik pada *class* itu sendiri ataupun pada *class* lainnya.

Dalam *class diagram* terdapat beberap simbol, beberapa simbol tersebut dapat dilihat pada tabel 2.2 dibawah ini:

Tabel 2.2 Simbol pada *Class Diagram*

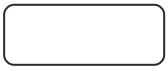
No	SIMBOL	PENJELASAN
1	Class1::**Class**	*Class*, digambarkan sebagai sebuah kotak yang terbagi atas 3 bagian. Bagian atas adalah bagian nama dari *class*. Bagian tengah mendefenisikan property/atribut *class*. Bagian akhir mendefenisikan *method-method* dari sebuah *class*.
2	* *	*Assosiation*, digunakan sebagai relasi antar dua kelas atau lebih.

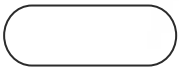
3		*Composition*, jika sebuah *class* tidak bisa berdiri sendiri dan harus merupakan bagian dari *class* yang lain, maka *class* tersebut memiliki relasi *composition* terhadap *class* tempat dia bergantung tersebut. Sebuah *Relationship composition* digambarkan sebagai garis dengan ujung berbentuk jajaran genjang berisi/solid.
4		*Dependency*, digunakan untuk menunjukkan operasi pada suatu *class* yang menggunakan *class* yang lain. Sebuah *dependency* dilambangkan sebagai sebuah panah bertitik-titik.

2.1.8 *Activity Diagram*

Activity Diagram menunjukan aktvitas sistem dalam bentuk kumpulan aksi-aksi, bagaimana masing masing aksi tersebut dimulai, keputusan yang mungkin tejadi hingga berakhirnya aksi. *Activity* diagram juga dapat menggambarkan proses lebih dari satu aksi dalam waktu bersamaan.”Diagram *Activity* adalah aktifitas-aktifitas, objek, *state*, transisi *state* dan *event (*Haviluddin *dalam* Suendri,2018). *Activity diagram* merupakan *state diagram* khusus, dimana sebagian besar *state* adalah *action* dan sebagian besar transisi di-*trigger* oleh selesainya *state* sebelumnya (*internal processing). Activity diagram* dapat digunakan untuk menjelaskan bisnis dan alur kerja operasional secara tahap demi tahap dari komponen suatu sistem. *Activity diagram* menunjukkan keseluruhan dari aliran *control*. Berikut ini ada beberapa simbol yang terdapat pada *activity* diagram, perhatikan pada Tabel 2.3 dibawah ini:

Tabel 2.3 Simbol pada *Activity Diagram*

No	SIMBOL	PENJELASAN
1		*Activity*, memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu

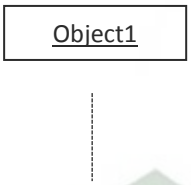
		sama lain.
2		*Action, state* dari sistem yang mencerninkan eksekusi dari suatu aksi.
3		*Initial State,* bagaimana objek dibentuk atau diawali.
4		*Final State,* bagaimana objek dibentuk dan diakhiri.
5		*Decision,* digunakan untuk menggambarkan suatu keputusan atau tindakan yang harus diambil pada kondisi tertentu.
6		*Control Flow,* menunjukkan bagaimana kendali suatu aktivitas terjadi pada aliran kerja dalam tindakan tertentu.

2.1.9 Sequence Diagram

Sequence diagram adalah *tool* yang sangat popular dalam pengembangan sistem informasi secara *object-oriented* untuk menampilkan interaksi antar objek (Nofriyadi Nurdam dalam Heriyanto,2018). Secara mudahnya *sequence* diagram adalah gambaran tahap demi tahap, termasuk kronologi (urutan) perubahan secara logis yang seharusnya dilakukan untuk menghasilkan sesuatu sesuai dengan *use case* diagram. Dalam *sequence diagram* terdapat beberapa simbol yang dapat dilihat pada tabel 2.4 dibawah ini:

Tabel 2.4 Simbol *Sequence Diagram*

No	SIMBOL	PENJELASAN
1		*Lifeline* mengindikasikan keberadaan sebuah *object* dalam basis waktu. Notasi untuk *Lifeline* adalah garis putus-putus *vertikal* yang ditarik dari sebuah *object.*

2		*Object* merupakan *instance* dari sebuah *class* dan dituliskan tersusun secara *horizontal.* Digambarkan sebagai sebuah *class* (kotak) dengan nama *object* didalamnya yang diawali dengan sebuah titik koma.
3	 *Actor1*	*Actor* juga dapat berkomunikasi dengan *object*, maka *actor* juga dapat diurutkan sebagai kolom. Simbol *actor* sama dengan simbol pada *Actor Case Diagram*.
4		*Activation* dinotasikan sebagai sebuah kotak segi empat yang digambar pada sebuah *lifeline.* Mengindikasikan sebuah objek yang akan melakukan sebuah aksi.

BAB III

METODOLOGI PENELITIAN

3.1 Metode Penelitian

Metode penelitian adalah ilmu yang mempelajari cara-cara melakukan pengamatan dengan pemikiran yang tepat secara terpadu melalui tahapan-tahapan yang disusun secara ilmiah mencari, menyusun serta menganalisis dan menyimpulkan data-data, sehingga dapat dipergunakan untuk menemukan, mengembangkan dan menguji kebenaran sesuatu pengetahuan berdasarkan bimbingan Tuhan. Metode juga merupakan analisis teoretis mengenai suatu metode.

1. Metode *Least Significant Bit* (LSB)

   Salah satu metode steganografi yang banyak digunakan adalah metode modifikasi LSB (*Least Significant Bit*). Metode modifikasi LSB tergolong metode yang menggunakan teknik substitusi. Metode LSB terutama digunakan untuk steganografi berbasis media (media-*based steganography*).

   Metode LSB menyembunyikan data rahasia dalam *bit-bit* tak signifikan (*least significant bit*) dari berkas wadah (*cover*). Pengubahan tersebut pada dasarnya memberikan pengaruh terhadap berkas wadah, tetapi karena perubahan yang terjadi sangat kecil, sehingga tidak tertangkap oleh indra manusia. Kenyataan inilah yang akhirnya dimanfaatkan sebagai teknik penyembunyian data atau pesan (steganografi).Perubahan yang tidak

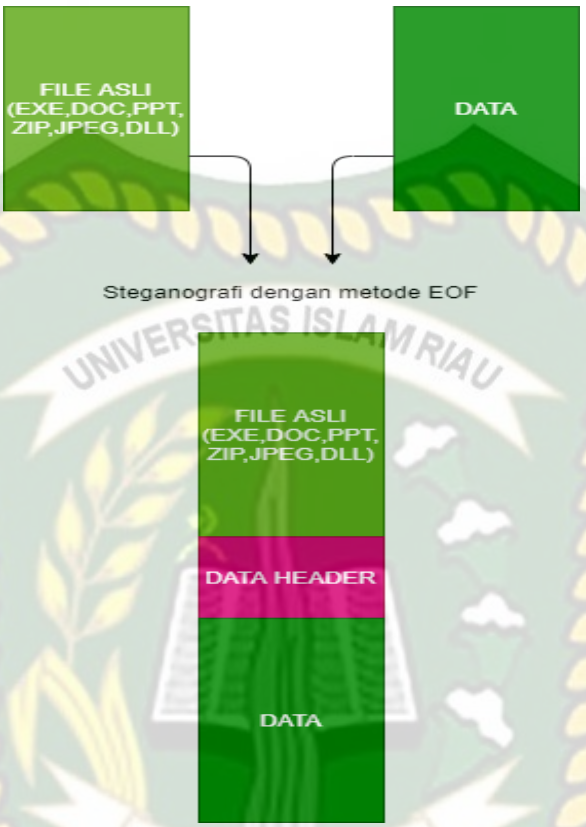
signifikan ini tidak akan tertangkap oleh indra manusia (jika media wadah adalah gambar, suara atau video).

Penggantian *pixel* tak signifikan juga dapat dilakukan secara tak terurut, bahkan hal ini dapat meningkatkan tingkat keamanan data (*imperceptability*). Disamping itu juga mungkin melakukan pengubahan *pixel* tidak pada bagian awal berkas wadah. Pengubahan *pixel* juga dapat dipilih mulai dari tengah, atau dari titik lain dari berkas wadah yang dimungkinkan untuk menyimpan seluruh informasi rahasia, tanpa menimbulkan permasalahan saat pengungkapan data. Meskipun metode LSB mudah diterapkan, akan tetapi steganografi dengan metode ini akan menghasilkan berkas stego yang mudah rusak (dirusak).

Steganografi dengan metode LSB juga hanya mampu menyimpan informasi dengan ukuran yang sangat terbatas. Misalnya suatu citra 24-*bit* (R=8-*bit*, G=8-bit, B=8-*bit*) digunakan sebagai wadah untuk menyimpan data berukuran 100 *bit*, jika masing – masing komponen warnanya (RGB) digunakan satu *pixel* untuk menyimpan informasi rahasia tersebut, maka setiap *pixel* disimpan 3 *bit* informasi, sehingga setidaknya dibutuhkan citra wadah berukuran 34 *pixel* atau setara 34 x 3 x 8 = 816 *bit* (8 kali lipat). Jadi suatu citra 24-*bit* jika digunakan untuk menyimpan informasi rahasia hanya mampu menampung informasi maksimum berukuran 1/8 dari ukuran citra penampung tersebut.

2. Metode End of File (EOF).

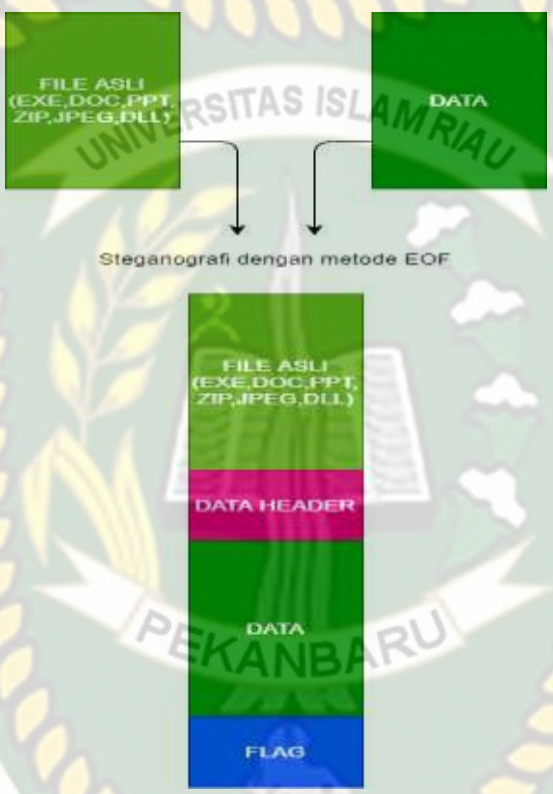
Metode End of File (EOF) Secara umum teknik steganografi menggunakan *redundant bits* sebagai tempat menyembunyikan pesan pada saat dilakukan kompresi data, dan kemudian menggunakan kelemahan indra manusia yang tidak *sensitive* sehingga pesan tersebut tidak ada perbedaan yang terlihat. Teknik EOF atau *End Of File* merupakan salah satu teknik yang digunakan dalam steganografi. Teknik ini digunakan dengan cara menambahkan data atau pesan rahasia pada akhir file. Perhitungan ukuran file yang telah disisipkan data sama dengan ukuran file sebelum disisipkan data ditambah ukuran data rahasia yang telah diubah menjadi *encoding file*. Dengan metode EOF, secara umum media steganografi (file yang akan disisipi data) memiliki struktur seperti gambar dibawah ini :



Gambar 3.1 Struktur File Steganografi dengan Metode *End of File*

Penanda data *header* atau *flag* akan kita letakkan di awal atau akhir file, di mana tidak ada *looping* yang digunakan untuk mencarinya. Pada beberapa file seperti exe dan zip, penempatan *flag* di awal file asli tidak akan menjadi masalah, namun untuk jenis file lain semisal JPG, BMP dan DOC, penempatan *flag* di awal file akan merusak file asli karena mengganggu isi file asli dan merusak CRC file tersebut. Kita akan menempatkannya di akhir file sehingga tidak membawa bencana meskipun

kita mengunakan berbagai jenis file. Ini juga sesuai dengan konsep *End Of File* pada steganografi ini :



Gambar 3.2 Struktur File Steganografi dengan Metode *End of File* disertai *flag*

3. Kedalaman *bit* merupakan faktor penting yang mempengaruhi kualitas gambar dan ukuran file. kedalaman *bit* mengacu pada jumlah informasi yang disimpan untuk setiap *pixel* dari suatu gambar . *bit* paling umum kedalaman untuk gambar 8-*bit* dan 24-*bit*. Kedalaman 8-*bit* dapat memuat gambar sebanyak 256 warna, namun kedalaman 24-*bit* dapat memuat gambar sebanyak 16.777.216 warna. Matematika perhitungan ukuran file (dalam *byte*s) dari suatu gambar dengan rumus berikut (Robert Reinhardt, Snow Dowd, 2009) :

Ukuran file = lebar (kolom) x tinggi (baris) x (kedalaman *bit*) / 8

4. MSE dan PSNR

Peak Signal to Noise Ratio (PSNR) adalah perbandingan antara nilai maksimum dari sinyal yang diukur dengan besarnya *noise* yang berpengaruh pada sinyal tersebut. PSNR dalam satuan desibel. Pada penelitian ini, PSNR digunakan untuk perbandingan kualitas citra *cover* sebelum dan sesudah disisipkan pesan. Untuk menentukan PSNR, terlebih dahulu harus ditentukan MSE (*Mean Square Error*). MSE adalah nilai *error* kuadrat rata-rata antara citra *cover* dengan citra tersteganografi, secara matematis dapat dirumuskan seperti pada persamaan berikut :

$$\mathbf{MSE}\ \frac{\mathbf{1}}{\boldsymbol{MM}} = \quad \sum_{Y=}^{\boldsymbol{M}} \mathbf{1} \sum \qquad\qquad {}_{Y=}^{\boldsymbol{N}}\mathbf{1}\ [\mathbf{I(x, y)} - \mathbf{I'(x, y)}]^2$$

Dimana :

MSE = Nilai MSE citra steganografi; M = Panjang citra *stego* (dalam *pixel*) N = Lebar citra *stego* (dalam *pixel*) ; I(x,y) = nilai *pixel* dari citra *cover* I'(x,y) = nilai *pixel* pada citra *stego*

Setelah diperoleh nilai MSE maka nilai PSNR dapat dihitung dari kuadrat nilai maksimum dibagi dengan MSE. Secara matematis, nilai PSNR dirumuskan seperti pada persamaan

PSNR = $\log_{10}(\frac{MAXI^2}{MSE})$

Dimana MSE adalah Nilai MSE; MAXi adalah nilai maksimum dari *pixel* citra Semakin rendah nilai MSE maka akan semakin baik, dan semakin besar nilai PSNR maka semakin baik kualitas citra steganografi.

5. Alat dan Bahan Penelitian Yang Digunakan

Adapun alat dan bahan penelitian ini adalah sebuah pendukung baik perangkat keras maupun perangkat lunak sehingga penelitian ini sesuai dengan tujuan dan manfaatnya. Berikut adalah alat dan bahan penelitian yang digunakan penulis untuk menganalisa dan merancang sistem.

3.2 Spesifikasi *Hardware* dan *Software*

Adapun spesifikasi perangkat keras dan perangkat lunak yang digunakan dalam melakukan penelitian ini dapat dilihat pada tabel 3.1 dibawah ini:

Tabel 3.1 Spesifikasi *Hardware* dan *Software*

No	*Hardware* dan *Software*	Spesifikasi	Fungsi

1	Laptop	• *Processor Intel Core* • Ram 4 GB • *Hardisk* 500 GB • 64-*bit Operating System*	• Sebagai media yang digunakan penulis melakukan beberapa tahapan dalam melakukan proses pengcodingan dan proses pengujian penelitian
2	Sistem Operasi	• *Microsoft Windows 10 Home Single Language* • *Sublime text* • *pyton*	• Sistem operasi yang digunakan penulis dalam melakukan penelitian • *Text* editor • Bahasa pemograman untuk proses enkripsi

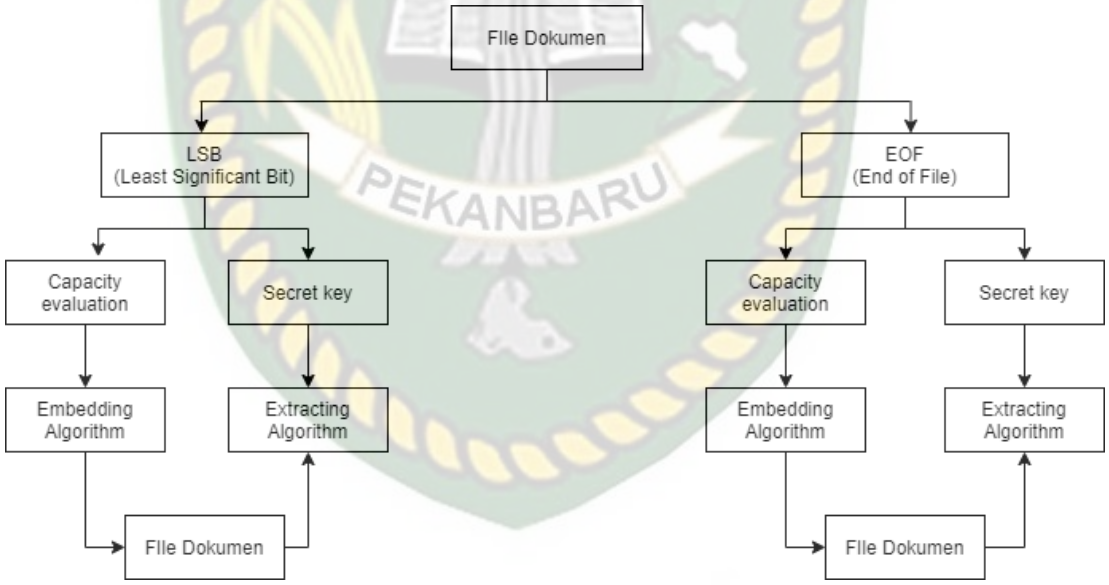
3.3 Usulan Skema Enkripsi

Pada penelitian ini dilakukan anilisa penyisipan file *text* ke dalam file dokumen menggunakan kombinasi alogritma EOF (*End of File*) dan LSB *(Least Significant Bit*). file dokumen yang digunakan adalah berformat pdf,doc dan lain lain, pada fase penyisipan untuk tempat penyisipan. Setiap dokumen bertindak sebagai media *cover* yang terdiri dari beberapa *frame* untuk tempat peyisipan.

Algoritma steganografi mempunyai metode yang bisa digunakan seperti LSB (*Least Significant Bit*) dan EOF (*End Of File*). Kedua algoritma ini mempunyai model yang sangat berbeda dalam proses penyamaran dan penyembunyian data (Gunawan, 2018) Lain dari itu, algoritma ini juga masih digunakan untuk pengembangan didalam ilmu steganografi itu sendiri agar dapat menghasilkan model-model terbaru dari algoritma steganografi. Dalam penelitian

sebelumnya dengan judul Pangamanan Acakan BISS Menggunakan Algoritma RSA (Gunawan, 2018), telah dijelaskan tentang bagaimana proses penyisipan teks kedalam sebuah video, lalu mengacaknya, akan tetapi akan terlihat perbedaan yang sangat jauh dari segi ukuran video yang sudah disisipkan oleh teks.

tersebut dapat dengan selamat sampai kepada si penerima/yang memiliki hak untuk membuka berkas dokumen tersebut. Dengan meningkatkan sistem keamanan dari berkas dokumen dapat membantu mengamankan data-data yang ada didalam berkas dokumen pada saat proses pengiriman data, sehingga berkas dokumen tersebut dapat dengan selamat sampai kepada si penerima/yang memiliki hak untuk membuka berkas dokumen tersebut.



Gambar 3.3 Usulan Skema Enkripsi

3.4 Pengembangan dan Perancangan Sistem

Pada perancangan Enkripsi Steganografi file dokumen menggunakan metode *Least Significant Bit* dan *End of File*. Proses Utama adalah antara lain Pertama proses Enkrispi dokumen, Kedua Ekstrasi file dokumen atau

pengembalian seperti semula atau mendekripsikan file dokumen yang telah di enkripsi sebelumnya.

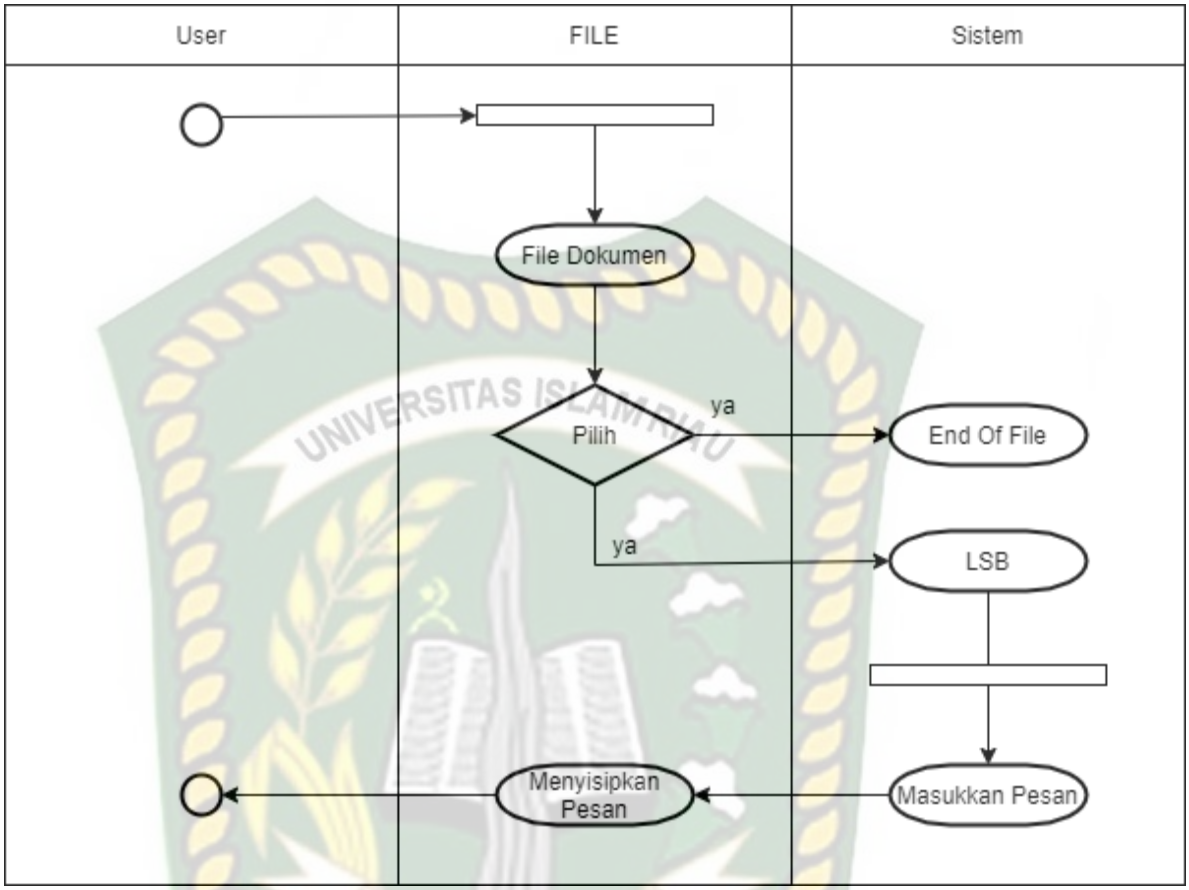
3.4.1 *Hirarchy Chart*

Hirarchy chart adalah diagram yang menggambarkan permasalahan kompleks yang kemudian diuraikan dalam beberapa elemen, berikut gambaran *hirarchy chart* pada sistem keamanan penyembunyian data didalam file dokumen menggunakan metode *Least significant bit* dan *End of File*, dapat dilihat pada gambar 3.4

Gambar 3.4 *Hirarchy Chart*

3.4.2 *Activity Diagram*

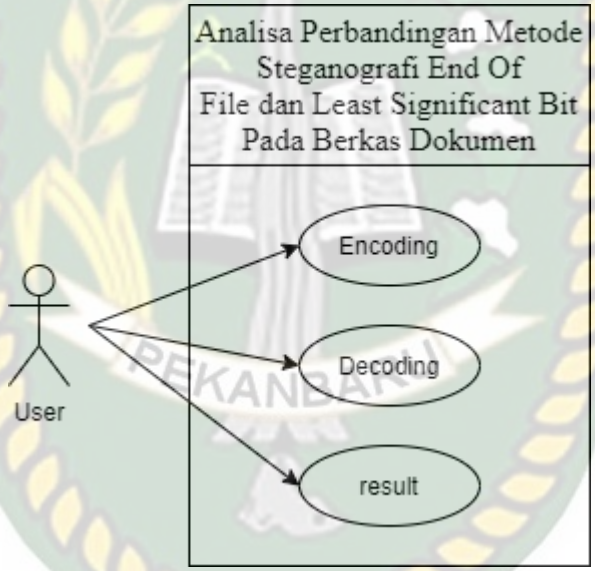
Activity Diagram menggambarkan berbagai alur aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* atau diagram aktivitas adalah representasi grafis dari alur kerja kegiatan bertahap dan tindakan dengan dukungan untuk pilihan, iterasi dan konkurensi. *Activity* diagram juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. Pada *Unified Modeling Language*, diagram aktivitas dapat digunakan untuk menjelaskan bisnis dan operasional langkah demi langkah alur kerja komponen di dalam sistem. Sebuah aktivitas dapat direalisasikan oleh satu *use case* atau lebih. Aktivitas menggambarkan proses yang berjalan, sementara *use case* menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas. *Activity Diagram* pada sistem ini dapat dilihat pada gambar 3.5



Gambar 3.5 *Activity Diagram Encoding* dan *Decoding*

3.4.3 *Use Case Diagram*

Pada perancangan aplikasi Analisa Perbandingan Metode Steganografi *End Of File* dan *Least Significant Bit* Pada Berkas Dokumen terdiri dari proses yang dilakukan oleh pengirim pesan yaitu, proses *encoding* pesan, dan *decoding*, *User Case* aplikasi yang akan dibangun dapat dilihat pada gambar 3.6 dibawah ini.



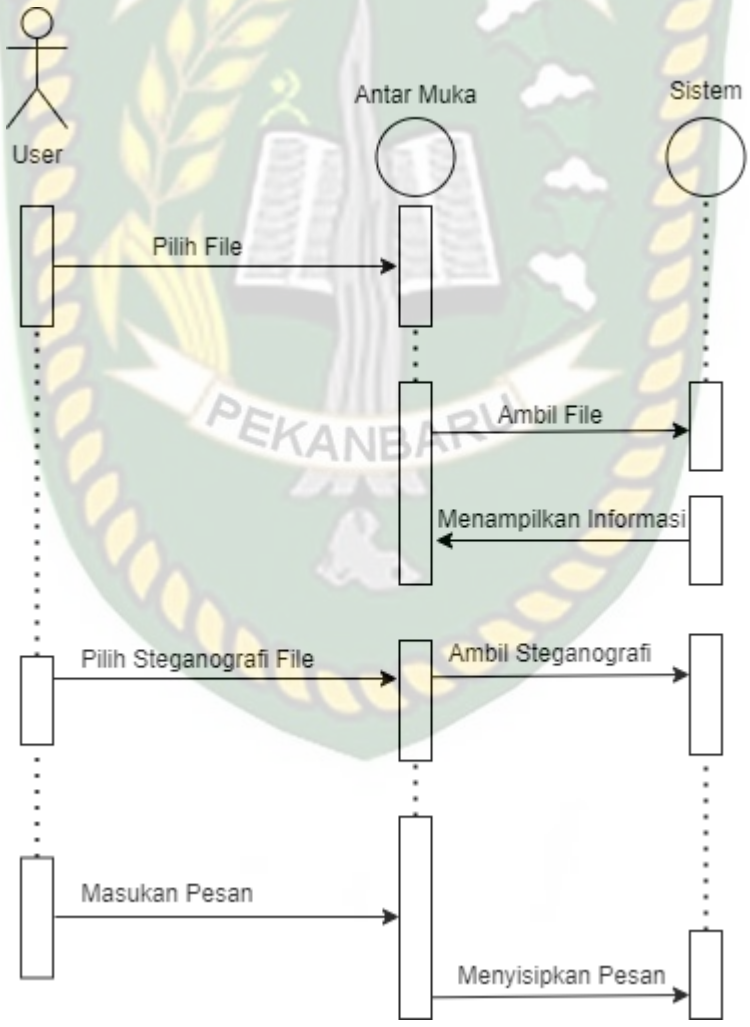
Gambar 3.6 *User Case Diagram*

Pada gambar 3.6 diatas dapat dilihat pada aplikasi yang akan dibangun terdiri dari 2 aktor *case Encoding* dan *Decoding* serta *result*.

3.4.4 *Sequence Diagram*

Sequence diagram digunakan untuk mengetahui tentang alur proses dan interaksi antara objek pada aplikasi yang akan dibangun. Dengan menggunakan *sequence diagram* kita dapat melihat bagaimana objek-objek bekerja. *Sequence*

diagram dapat menampilkan bagaimana sistem merespon setiap kejadian atau permintaan dari *user*, dapat mempertahankan integritas *internal*, bagaimana data dipindahkan ke *user interface* dan bagaimana objek-objek diciptakan dan dimanipulasi, *Sequence Diagram* pada proses pengirim pesan dapat dilihat pada gambar 3.7 dibawah ini :



Gambar 3.7 *Sequence Diagram*

Berdasarkan gambar 3.7 diatas pengirim membuat pesan dapat dilihat langkah-langkah yang dilakukan pengirim pesan mulai dari menjalankan aplikasi sampai dengan melakukan proses penyandian, penyembunyian dan mengenkripsi isi pesan pada File Dokumen, kemudian menyimpan *stego*.

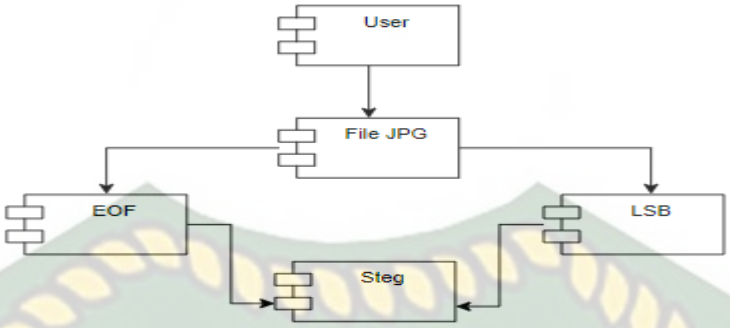
3.4.5 *Class Diagram*

Class Diagram menggambarkan struktur dan deskripsi *class*, *package*, dan objek yang saling terhubung. *Class Diagram* yang dijelaskan pada analisa ini adalah *class diagram* pada aplikasi yang akan dibangun, seperti gambar 3.8 dibawah ini :

Gambar 3.8 *Class Diagram*

3.4.6 *Component Diagram*

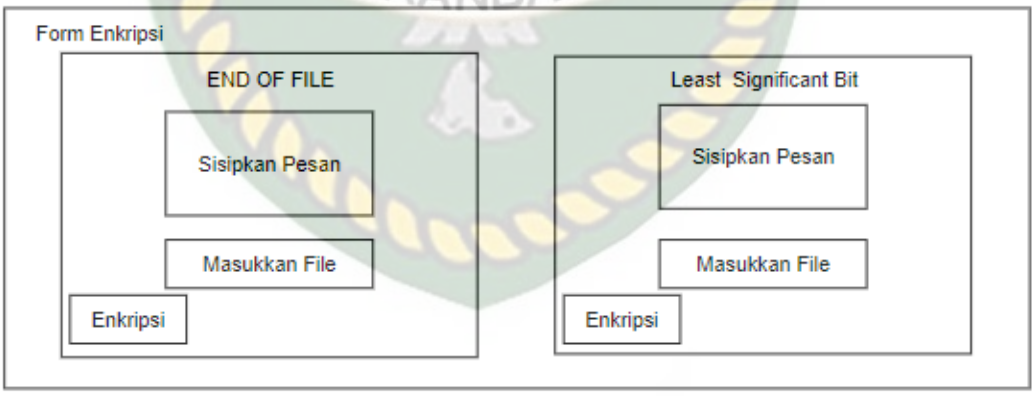
Component diagram adalah diagram UML yang menampilkan komponen dalam sistem dan hubungan antara mereka. Dibawah ini adalah gambar skema *component diagram*. Dibawah ini adalah gambar skema *component diagram*. Pada component diagram dibawah dijelaskan *User* yang beinteraksi dengan file dokumen juga menjadi *controller interface*, file dokumen dipilih metode yang digunakan EOF atau LSB sehingga dapat mengeluarkan hasil steganografi menjadi format yang diinginkan.



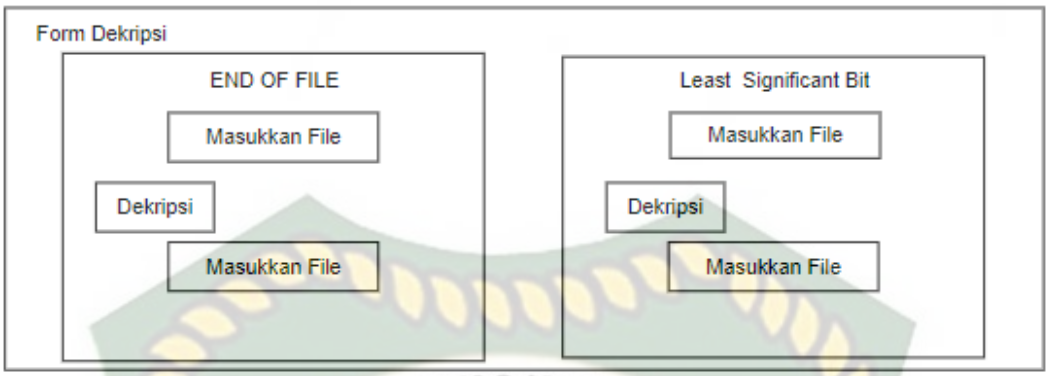
Gambar 3.9 *Component Diagram*

3.4.7 Perancangan Input Enkripsi dan Dekripsi

Desain *input* merupakan perancangan desain masukan dari pengguna kepada sistem. Desain *input encoding* ini merupakan bentuk tampilan yang digunakan untuk melakukan proses *input* pesan dan File Dokumen. Tampilan *input* enkripsi dan dekripsi dapat dilihat pada gambar 3.10 dan 3.11 dibawah



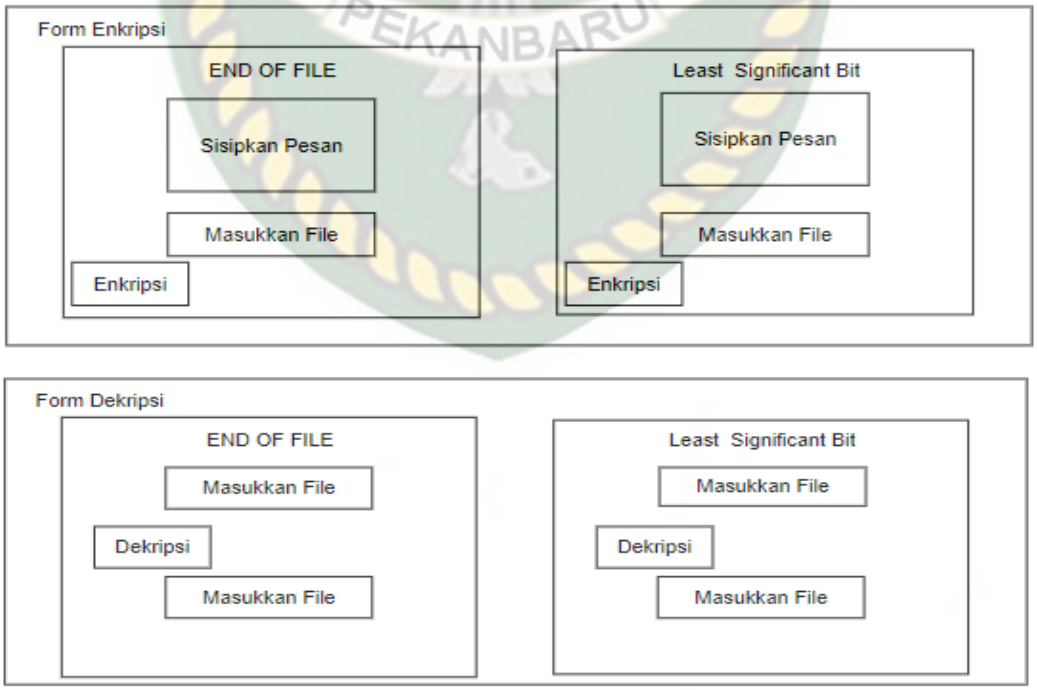
Gambar 3.10 Desain *Input* Enkripsi



Gambar 3.11 Desain *Input* Dekripsi

3.4.8 Desain *Interface* Halaman Utama

Pada halaman utama menampilkan halaman awal pada sistem pertama kali dijalankan. Adapun tampilan halaman utama dapat dilihat pada Gambar 3.12 dibawah ini.



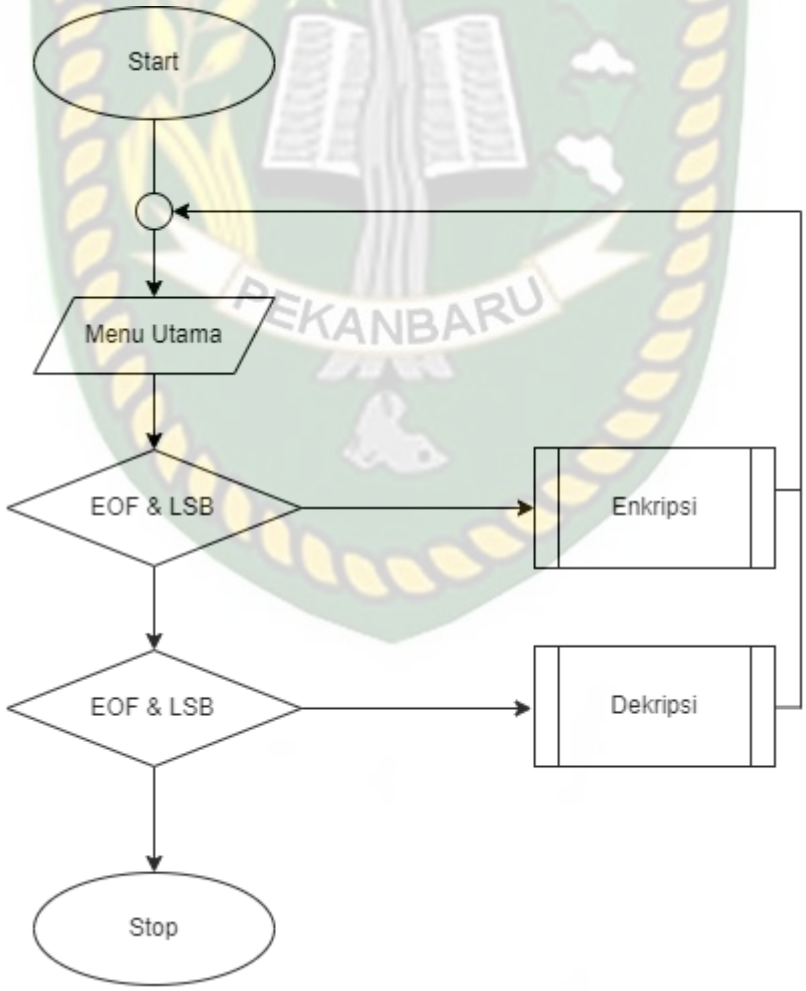
Gambar 3.12 Halaman Utama

3.4.9 Perancangan Logika Program

Perancangan logika program memberikan gambaran bagaimana sistem bekerja mulai dari proses *input* sampai dengan proses *output*. Dan memberikan gambaran kinerja sistem yang terstruktur dan sistematis.

3.4.10 *Flowchart* Menu Utama

Pada *Flowchart* Menu Utama yang terlihat saat program dijalankan adalah menu utama akan menampilkan menu Enkripsi dan Dekripsi. Seperti yang dapat dilihat pada gambar 3.13 dibawah ini



Gambar 3.13 *Flowchart* Halaman Utama

BAB IV
HASIL DAN PEMBAHASAN

Dalam pembuatan aplikasi yang telah dirancang dan dibangun maka dilakukan pengujian terlebih dahulu, pengujian yang dilakukan untuk mengetahui hasil yang diberikan oleh aplikasi Analisa Perbandingan Steganografi Berkas Dokumen Dengan Metode *End Of File* Dan LSB. Pengujian yang akan dilakukan pada aplikasi ini dengan metode *black box.*

4.1 Pengujian *Black Box*

Pengujian *black box* (*Black Box Testing*) adalah salah satu metode pengujian perangkat lunak yang berfokus pada sisi fungsionalitas sistem, khususnya pada input dan output, apakah sistem telah sesuai dengan yang diharapkan atau belum. Maka hasil dari pengujian *black box.*

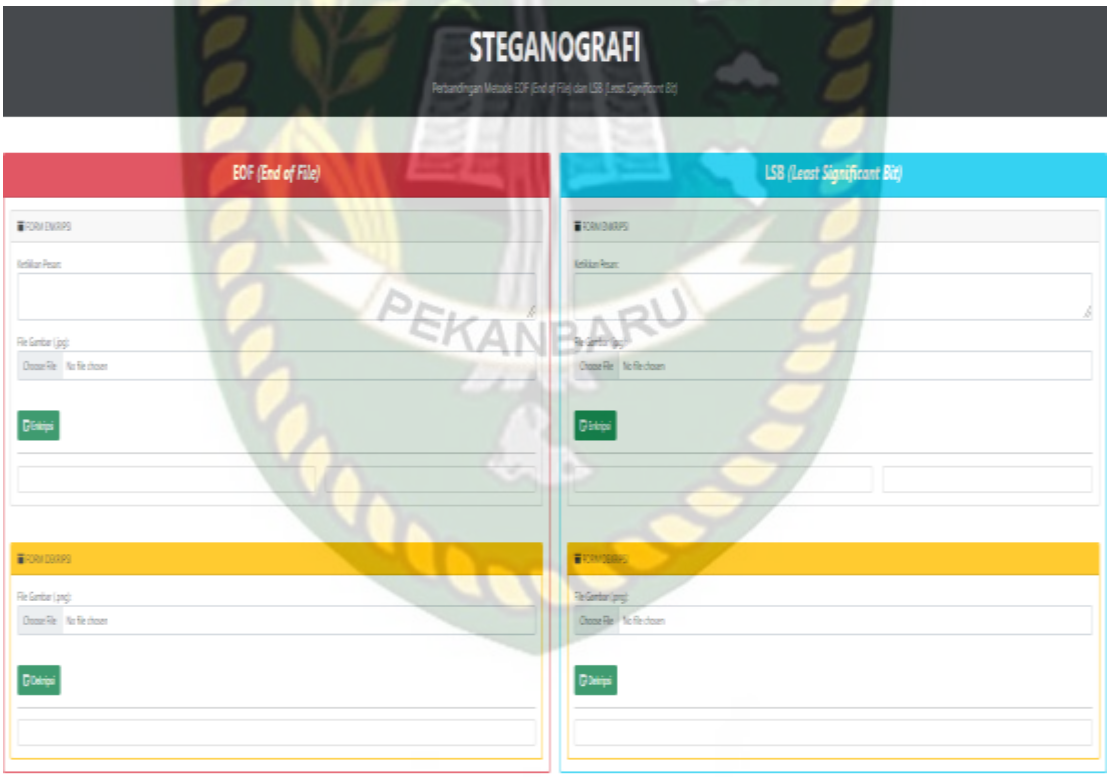
Tabel 4.1 Pengujian *Black box*

Deskripsi	Prosedur Pengujian	Masukan	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
Enkripsi	Memasukkan file baru yang akan diuji	Sisipan Pesan	Memunculkan hasil enkripsi	Sesuai harapan	Berhasil
Deskripsi	Memasukkan stego file yang telah di lakukan sebelumnya	Masukkan file yang telah di enkripsi	Memunculkan hasil Deskripsi	Sesuai harapan	Berhasil

4.2 Penjelasan Sistem

4.2.1 Form Halaman Utama Enkripsi Dan Dekripsi

Pada halaman menu utama terdapat menu-menu pilihan yang memiliki fungsi masing-masing, yaitu menu Enkripsi dan Dekripsi. Menu Enkripsi merupakan tempat melakukan pengenkripsian pesan dengan File Jpg. Menu Dekripsi merupakan tempat pemisah atau pengekstrasian pesan yang telah di enkrpsi dengan File Png. Untuk lebih jelasnya dapat dilihat pada gambar 4.1



Gambar 4.1 Tampilan Halaman Utama Program

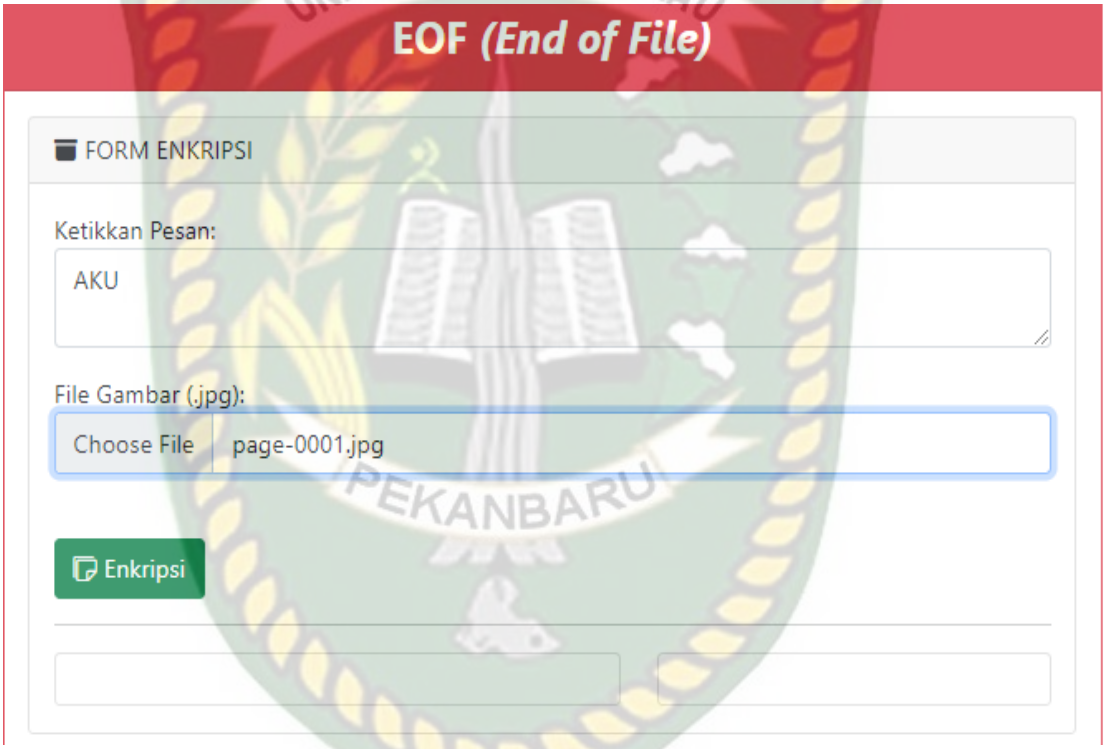
4.3 Hasil Analisis

Program yang sudah siap dirancang selanjutnya dilakukan implementasi untuk melakukan proses penyisipan dengan kedua metode yang telah dirancang

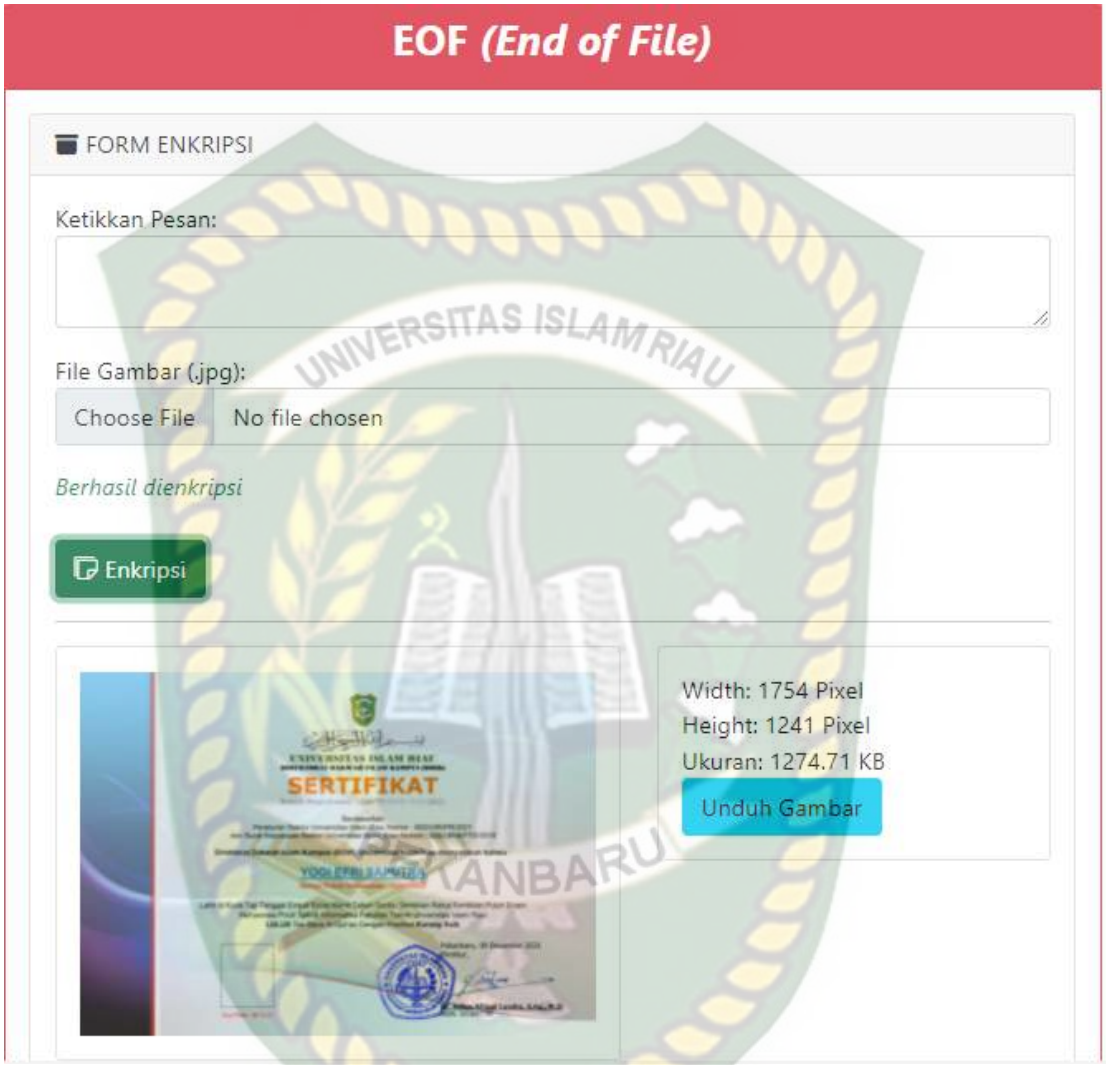
4.3.1 Hasil Steganografi EOF

1. Enkripsi EOF

Tampilan hasil steganografi EOF adalah program untuk melakukan penyisipan file *image* dengan algoritma EOF, dan hasil enkripsi. Dapat dilihat pada gambar 4.2 dan 4.3.



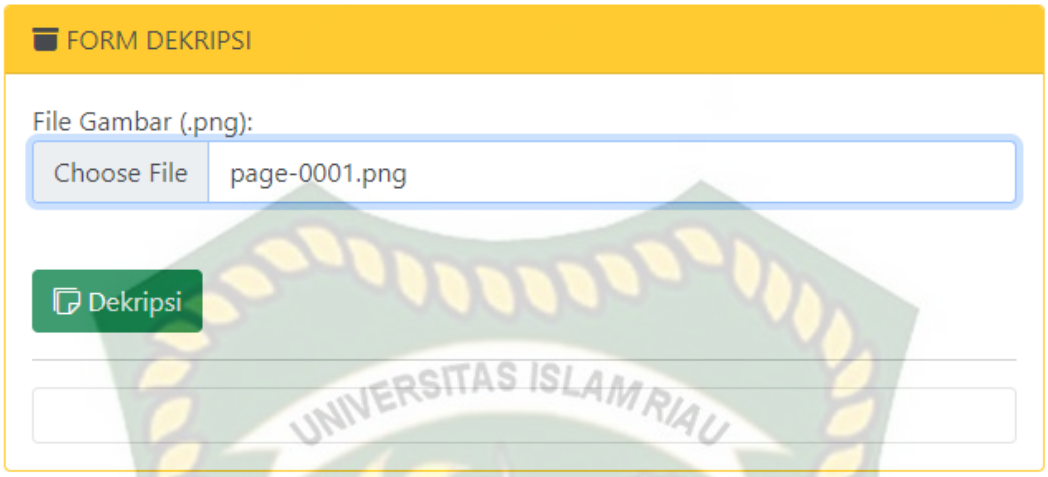
Gambar 4.2 Tampilan Menyisipkan Pesan dengan metode EOF



Gambar 4.3 Tampilan Hasil Enkripsi dengan metode EOF

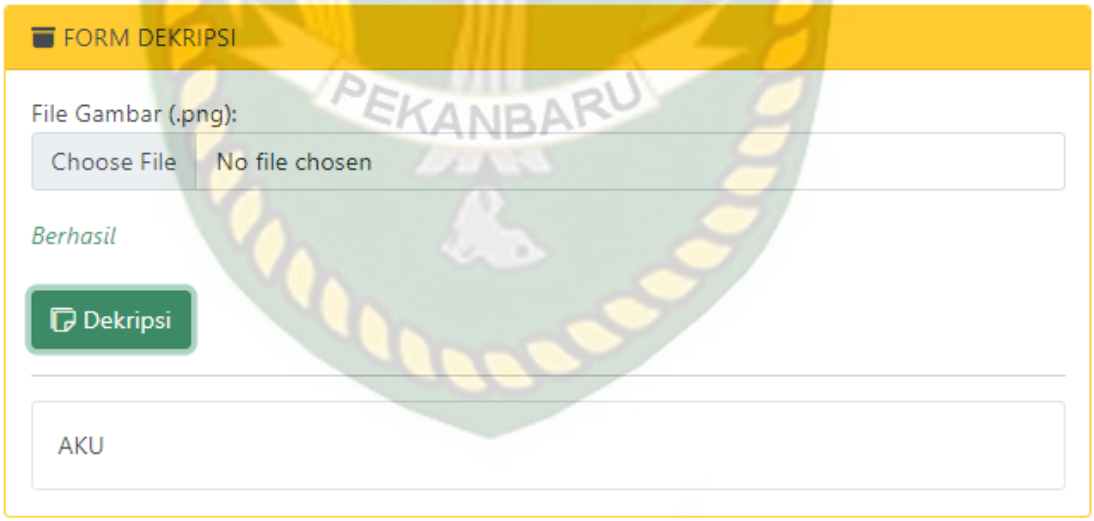
2. Dekripsi EOF

Tampilan hasil steganografi EOF adalah program untuk melihat penyisipan file *image* dengan algoritma EOF, dan hasil dekripsi. Dari unduhan file di dipilih file format png dapat dilihat pada gambar 4.4



Gambar 4.4 Tampilan memilih file dengan metode EOF

Dari unduhan file yang sudah di enkripsi sebelumnya dapat dilihat hasil deskripsi pada gambar 4.5

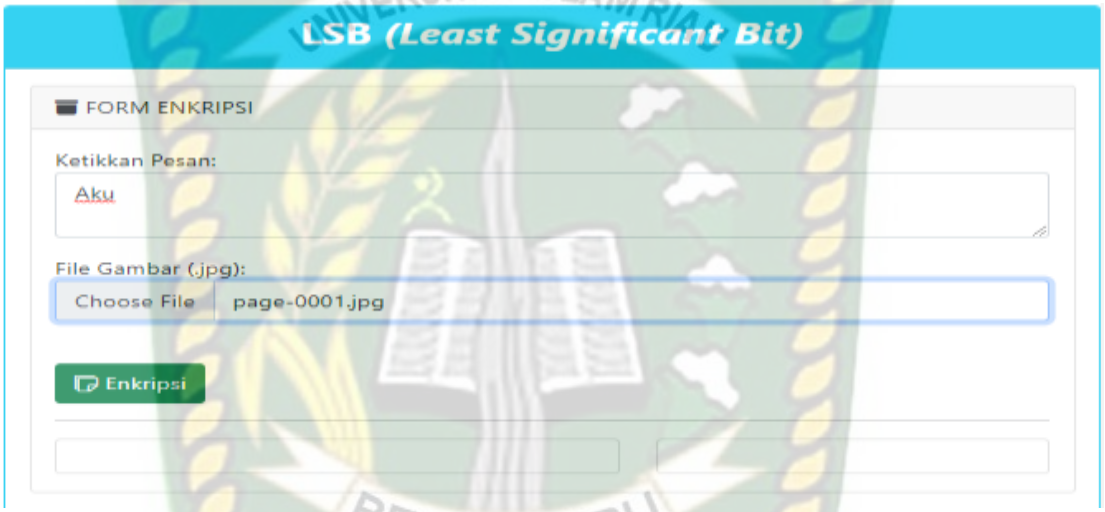


Gambar 4.5 Tampilan Hasil Dekripsi dengan metode EOF

4.3.2 Hasil Steganografi LSB

1. Enkripsi LSB

Tampilan hasil steganografi LSB adalah program untuk melakukan penyisipan file *image* dengan algoritma LSB, dan hasil enkripsi. Dapat dilihat pada gambar 4.6 dan 4.7

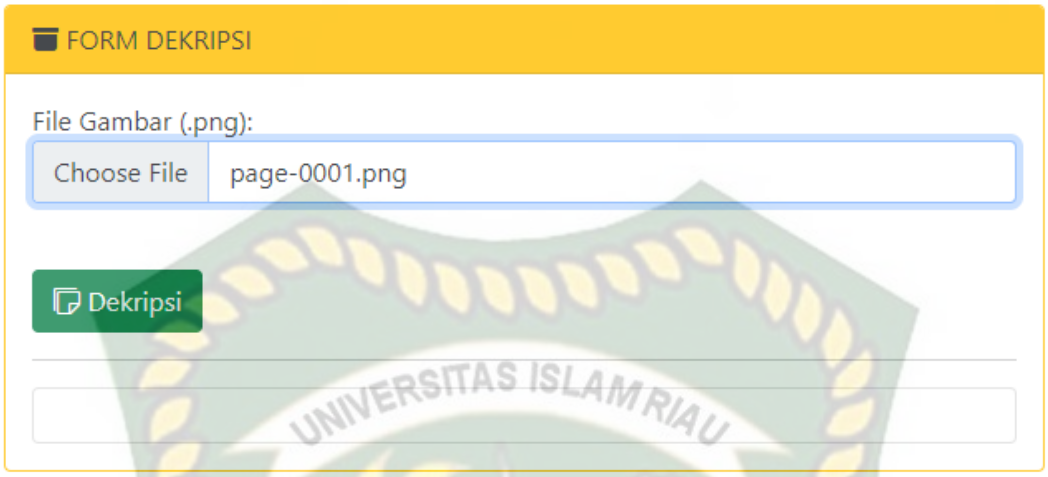


Gambar 4.6 Tampilan Menyisipkan Pesan dengan metode LSB

Gambar 4.7 Menampilkan Hasil Enkrpsi dengan metode LSB

2. Dekripsi LSB

Tampilan hasil steganografi LSB adalah program untuk melihat penyisipan file *image* dengan algoritma LSB, dan hasil dekripsi. Dari unduhan file di dipilih file format png dapat dilihat pada gambar 4.8



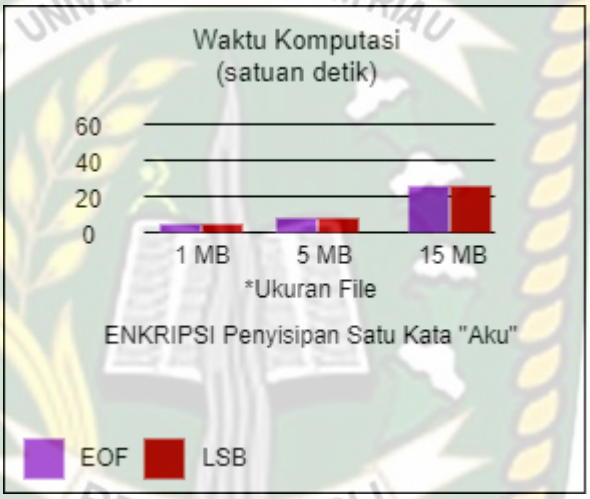
Gambar 4.8 Tampilan memilih file dengan metode LSB

Dari unduhan file yang sudah di enkripsi sebelumnya dapat dilihat pada gambar 4.9

Gambar 4.9 Tampilan Hasil Dekripsi dengan metode LSB

4.3.3 Pengujian Perbandingan Enkripsi dan Dekripsi pada metode EOF dan LSB

Pada pengujian Enkripsi Pada metode EOF dan LSB dibawah ini akan melakukan pengujian perbandingan dengan menentukan besar file jika disisipkan satu buah kata dan menghasilkan waktu enkripsi dari dua metode tersebut bisa dilihat pada gambar 4.10



Gambar 4.10 Perbandingan Enkripsi penyisipan satu kata berdasarkan ukuran file

Tabel 4.2 Hasil Perbandingan Metode LSB dan EOF

Metode LSB	Metode EOF
1. Ukuran *file t*idak ada perubahan saat penambahan *pixel*.	1. Ukuran *file* mengalami peberubah akibat penambahan *pixel*.
2. Jumlah *pixel* citra tetap pada setiap foto yang di enksripsikan.	2. Jumlah *pixsel* akan bertambah pada setiap foto yang di enkripsikan.
3. Lebih banyak menapung penyisipan kata.	3. Memiliki batasan pada saat penyisipan kata maksimal 50 kata jika lebih maka akan lama saat proses enkripsi.
4. Saat kata melampaui 50 kata tidak berpengaruh saat proses enkripsi.	4. Untuk kata yang lebih dari 50 memakan waktu 15 menit

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil dari penelitian diatas dapat disimpulkan beberapa hal seperti berikut :

1. Setelah dilakukan pengujian kedua metode EOF dan LSB maka dapat disimpulkan kedua metode ini hampir sama dalam mengenkripsi ataupun dekripsi file yang ada.
2. Keunggulan lebih banyak didapatkan pada metode LSB dibandingkan dengan metode EOF dikarenakan citra setelah disisipkan pesan hanya mengalami sedikit penurunan kualitas yang tidak begitu berpengaruh secara signifikan bila dilihat oleh mata manusia, sedangkan metode EOF mengalami perubahan yang signifikan pada ukuran citra, sehingga untuk metode penyisipan pesan pada gambar lebih baik jika digunakan menggunakan metode LSB.

5.2 Saran

Dari hasi penelitian ini, adapun beberapa saran yang dibutuhkan adalah :

1. Untuk menyempurnakan hasil dan menambahkan file *image* dengan ekstensi selain yang digunakan pada kasus ini.
2. Semoga kedepannya hasil analisa perbandingan metode EOF dan LSB memberikan perbandingan yang memperlihatkan keunggulan dari tiap metode yang ada sehingga kita dapat menggunakan metode steganografi yang tepat.

DAFTAR PUSTAKA

(Cahyadi. T. 2012). Implementasi Vigenere Cipher Sebagai Pengaman Pada Proses Deskripsi Steganografi Least Significant Bit. Jakarta.

(Darmadi dkk. 2004). Brand Equity Ten Strategi Memimpin Pasar. Jakarta: Gramedia Pustaka Utama.

(Hidayat, A. 2009). Metode Penelitian Keperawatan dan Tekhnik. Analisis Data. Jakarta: Salemba Medika.

(Indra Gunawan, 2019). Modifikasi Keamanan File dengan Algoritma Hill Cipher Untuk Mengantisipasi Dari Serangan Brute Force, Sumatera Utara.

(Indriyono, V. 2016). Klasifikasi Jenis Buku Berdasarkan Judul dan Sinopsis Menggunakan Metode Naïve Bayes Classifier (Studi Kasus : STMIK Kadiri).

(Nofriyadi Nurdam, 2014). Sequence Diagram Sebagai Perkakas Perancangan Antarmuka Pemakai. B2TKS–BPPT Jakarta Indonesia. Ultimatics. Vol. VI. No. 1. Juni 2014. ISSN: 2085-4552.

(Shalahuddin, M. 2015) Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek. Bandung: Informatika. Available at: https://scholar.google.co.id/scholar?hl=id&as_sdt=0,5&cluster=27834975078094069 89.

(Siti Fatima, 2015). Diagram Unifued Modelling Launguage (UML). Sumatera Utara.

(Suendri.2018).Implementasi Diagram UML (Unified Modelling Language) Pada Perancangan Sistem Informasi Remunerasi Dosen Dengan Database

Oracle (Studi Kasus: UIN Sumatera Utara Medan). Medan: ALGORITMA: Jurnal Ilmu Komputer dan Informatika.Vol.3.No.1