



**PERANCANGAN TEKNOLOGI STEGANOGRAFI PADA
PENGOLAHAN CITRA DIGITAL MENGGUNAKAN
METODE END OF FILE**

Disusun dan Diajukan untuk Memenuhi Persyaratan Ujian Akhir Memperoleh
Gelar Sarjana Komputer pada Fakultas Sains dan Teknologi
Universitas Pembangunan Panca Budi
Medan

SKRIPSI

OLEH

NAMA : FALDO
NPM : 1724370890
PROGRAM STUDI : SISTEM KOMPUTER

**FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS PEMBANGUNAN PANCA BUDI
MEDAN
2019**

ABSTRAK

FALDO **PERANCANGAN TEKNOLOGI STEGANOGRAFI PADA** **PENGOLAHAN CITRA DIGITAL MENGGUNAKAN** **METODE END OF FILE**

Perkembangan teknologi informasi saat ini, semakin memudahkan para pelaku kejahatan komputer, dengan menyalahgunakan teknologi tersebut untuk mendukung kegiatannya, dimana aktivitas mereka sangat mengganggu privasi seseorang. di skripsi ini penyisipan pesan teks dengan Metode End Of File. Oleh karena itu diperlukan sebuah sistem atau aplikasi yang aman sehingga dapat mempersulit para pelaku kejahatan komputer untuk melakukan aktivitasnya, dan membantu para pengguna teknologi dalam hal pengamanan data yang diakses tersebut. Untuk mempersulit para pelaku kejahatan komputer maka penulis membuat dengan metode Metode End Of File, yang diharapkan mampu menambah keamanan sebuah pesan teks rahasia. Untuk meningkatkan keamanan data yang akan disimpan, data yang disimpan juga dienkripsi terlebih dahulu.

Kata Kunci : *Metode End Of File, Enkripsi, Dekripsi*

DAFTAR ISI

	Halaman
KATA PENGANTAR.....	i
DAFTAR ISI.....	iii
DAFTAR GAMBAR.....	v
DAFTAR TABEL	vi
DAFTAR LAMPIRAN	vii
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Perumusan Masalah	2
1.3 Batasan Masalah	3
1.4 Tujuan Penelitian	3
1.5 Manfaat Penelitian	3
1.6 Metode Penelitian	4
1.7 Sistematika Penulisan	4
BAB II LANDASAN TEORI	7
2.1 Citra Digital	7
a. Citra Biner	8
b. Citra Warna	9
2.2 Pengolahan Citra Digital	9
a. Defini Pengolahan Citra	9
b. Tujuan Pengolahan Citra Digital	10
2.3 Metode EOF (End Of File)	10
2.4 UML (Unified Modeling Language)	11
1. Activity Diagram	12
2. Sequence Diagram	14
3. Use Case Diagram	15
2.5 Bahasa Pemrograman	18
2.6 Visual Basic. Net	19
2.7 Microsoft Visual Studi.....	22
BAB III METODE PENELITIAN	23
3.1 Tahap Penelitian	23
3.2 Metode Pengumpulan Data	24
3.3 Analisis Permasalahan Yang Berjalan	24
3.4 Analisa Kelemahan Yang Berjalan	25
3.5 Solusi Pemecahan Masalah	25
3.6 Analisa Proses Yang Berjalan	27

	1. Proses Embedding	27
	2. Proses Extraction	29
3.7	Perancangan Berorientasi Objek	31
	1. Use Case Diagram	31
	2. Activity Diagram	32
	3. Sequence Diagram	33
3.8	Perancangan Antarmuka	34
	1. Halaman Steganografi dengan LSB	34
BAB IV	HASIL DAN PEMBAHASAN	35
4.1	Implementasi Sistem.....	35
	4.1.1 Analisa Kebutuhan Sistem	35
	1. Analisis Perangkat Keras (Hardware)	35
	2. Analisis Perangkat Lunak (Software)	36
	4.1.2 Algoritma Least Significant Bit (LSB)	36
	1. Proses Penyisipan Data Teks (Embedded)	36
	2. Proses Mengambil Data Teks (Extraction)	37
4.2	Implementasi Sistem	38
	4.2.1 Tampilan Halaman Steganografi	39
	4.2.2 Tampilan Cari Gambar	40
	4.2.3 Tampilan Penyembunyian Pesan Text	41
	4.2.4 Tampilan Menampilkan Pesan Text	42
4.3	Pengujian Sistem	43
	4.3.1 Rencana Pengujian Alpha	44
	4.3.2 Rencana Pengujian Beta.....	44
	4.3.3 Kesimpulan dan Hasil Pengujian Alpha Beta	47
BAB V	PENUTUP	48
5.1	Kesimpulan.....	48
5.2	Saran	48

DAFTAR PUSTAKA
BIOGRAFI PENULIS
LAMPIRAN

DAFTAR GAMBAR

No	Judul	Hal
2.1	Ilustrasi Citra Digital	7
2.2	Contoh Citra Biner Berukuran 2x2 Pixel	8
3.1	Tahap Penelitian	23
3.2	Analisis Permasalahan yang Berjalan.....	25
3.3	Worksheet.xlsx	28
3.4	File Excel	30
3.5	Use Case Diagram	32
3.6	Activity Diagram	32
3.7	Sequence Diagram	33
3.8	Rancangan Form Menu Utama	34
4.1	Tampilan Halaman Menu Utama	39
4.2	Tampilan Cari Gambar	40
4.3	Tampilan Penyembunyian Pesan Text	41
4.4	Tampilan Gambar Yang Tersimpan Text.....	40
4.5	Tampilan Penyembunyian Pesan Text	45

DAFTAR TABEL

No	Judul	Hal
2.1	Notasi <i>Activity</i> Diagram	12
2.2	Simbol Sequence Diagram	15
2.3	Simbol Use Case Diagram	16
3.1	Tabel Perencanaan Rancangan	26
3.2	Nilai biner teks AKU	28
3.3	Tabel Biner Ms. Excel	28
3.4	Tabel Biner Ms.Excel yang berisi pesan rahasia.....	29
3.5	Tabel Biner Ms.Excel yang berisi pesan rahasia	30
3.6	Tabel biner pesan rahasia yang disisipkan.....	31
4.1	Rencana Pengujian Cari Gambar.....	44
4.2	Rencana Pengujian Pengguna (User)	44
4.3	Pengujian Input Gambar	45
4.4	Pengujian Input Pesan	45
4.5	Pengujian Input Password	46
4.6	Pengujian Menampilkan Pesan	46
4.7	Kesimpulan Pengujian Sistem	47

BAB I

PENDAHULUAN

1.1 Latar Belakang

Steganografi adalah seni dan ilmu untuk menyembunyikan pesan rahasia di dalam pesan lain sehingga keberadaan pesan rahasia tersebut tidak dapat diketahui. Berbeda dengan kriptografi yang merahasiakan makna pesan namun keberadaan pesan tetap ada, steganografi merahasiakan dengan menutupi atau menyembunyikan pesan (Harjo, 2016). Steganografi pada dunia digitalisasi telah banyak diterapkan untuk mengirimkan pesan atau informasi rahasia. Tidak hanya itu steganografi juga sering digunakan untuk kegiatan pengarsipan dimana data-data digital di sembunyikan kedalam berkas-berkas digital lainnya yang lebih umum sehingga tidak menarik perhatian pihak-pihak yang ingin mencuri informasi.

Steganografi seiring dengan perkembangannya telah melahirkan berbagai teknik dan metode yang berbeda – beda. Pada berkas digital multimedia seperti citra digital, metode steganografi yang paling umum digunakan adalah metode *Least Significant Bit* atau EOF (Singh & Singh, 2015). Metode End Of File (EOF) merupakan salah satu metode steganografi dalam teknik domain spasial. Metode EOF merubah nilai komponen warna bit terakhir dengan bit pesan yang akan disembunyikan sehingga menghasilkan citra yang mirip dengan aslinya. Metode ini dapat dikembangkan pada penyembunyian pesan rahasia.

Berdasarkan penjabaran diatas maka penulis bermaksud untuk melakukan penelitian dengan mengembangkan aplikasi pengamanan penyembunyian pesan teks pada citra digital menggunakan metode EOF dengan judul **“Perancangan Teknologi Steganografi Pada Pengolahan Citra Digital Menggunakan Metode End Of File”**.

1.2 Perumusan Masalah

Dalam pelaksanaan penelitian ini, adapun masalah yang diangkat, dibahas, dan diselesaikan adalah:

- a. Bagaimana menerapkan implementasi metode EOF pada penyembunyian pesan teks pada citra digital ?.
- b. Bagaimana menganalisis kinerja dari metode EOF dilihat dari keberhasilan penyisipan dan ekstraksi pesan pada citra digital?

1.3 Batasan Masalah

Agar tidak memperluas materi penulisan maka batasan-batasan dan ruang lingkup penulisan antara lain adalah:

- a. Format Citra yang digunakan berupa BMP, JPG dan PNG.
- b. Ukuran Citra yang digunakan 100 x 100 sampai dengan 1024 x 768.
- c. Pesan yang digunakan merupakan pesan teks (ASCII).

1.4 Tujuan Penelitian

Berdasarkan perumusan dan batasan masalah yang tertera diatas, ada pun tujuan dari penelitian ini, antara lain :

- a. Untuk mengetahui mekanisme dari metode *EOF*.
- b. Untuk meneliti metode *EOF* pada penyembunyian pesan teks pada citra digital.
- c. Membangun aplikasi perangkat lunak komputer yang dapat digunakan untuk pengujian dan implementasi steganografi pesan teks pada citra digital.

1.5 Manfaat Penelitian

Dari penjabaran di atas ada pula manfaat yang diberikan, adalah :

- a. Memahami bagaimana cara kerja metode *EOF* pada penyembunyian pesan teks pada citra digital.
- b. Membantu pengguna dalam memahami dan menggunakan aplikasi steganografi *EOF*.

1.6 Metode Penelitian

Metode pengumpulan data yang digunakan dalam pembahasan masalah ini adalah sebagai berikut:

- a. Studi Literatur : yaitu tahap pengumpulan data dengan cara mengumpulkan literatur, jurnal, paper, dan buku-buku yang berkaitan dengan judul penelitian, serta mencari informasi dari berbagai sumber di internet untuk mengetahui perkembangan terbaru dari data yang diambil
- b. Metode Pengembangan: tahap pengembangan aplikasi merupakan tahap pembuatan sistem yang dilakukan dengan menerapkan konsep steganografi pada citra digital yang digunakan untuk menunjang kelengkapan tugas akhir.
- c. Metode Analisis: melakukan analisis hasil uji coba program kepada para pengguna dan pemilik pesan pada saat sistem telah dikembangkan.

1.7 Sistematika Penulisan

Sistematika penulisan dari skripsi ini terdiri dari beberapa bagian utama sebagai berikut:

BAB I PENDAHULUAN

Bab ini akan menjelaskan mengenai latar belakang pemilihan judul skripsi “*Implementasi Steganografi EOF Pada Penyembunyian Pesan Teks Pada Citra Digital*”, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metode penelitian dan sistematika penulisan.

BAB II LANDASAN TEORI

Bab ini berisi dasar teori-teori yang digunakan dalam analisis, perancangan dan implementasi skripsi.

BAB III ANALISA DAN PERANCANGAN SISTEM

Bab ini berisi analisis terhadap fokus permasalahan penelitian dan perancangan terhadap sistem steganografi pesan teks pada citra digital. Kegiatan analisis terdiri dari analisis kebutuhan perangkat keras, perangkat lunak dan analisis algoritma EOF. Perancangan dilakukan dengan melakukan perancangan *use case diagram*, *flowchart* dan antarmuka sistem.

BAB IV IMPLEMENTASI DAN PENGUJIAN SISTEM

Bab ini berisi teknik implementasi dari perancangan yang telah dibuat dan pengujian terhadap implementasi. Pengujian dilakukan untuk membuktikan perangkat lunak dapat berjalan sesuai dengan spesifikasi yang telah ditentukan di tahapan analisis.

BAB V PENUTUP

Bab ini akan memuat kesimpulan isi dari keseluruhan uraian bab-bab sebelumnya dan saran-saran dari hasil yang diperoleh yang diharapkan dapat bermanfaat untuk pengembangan selanjutnya.

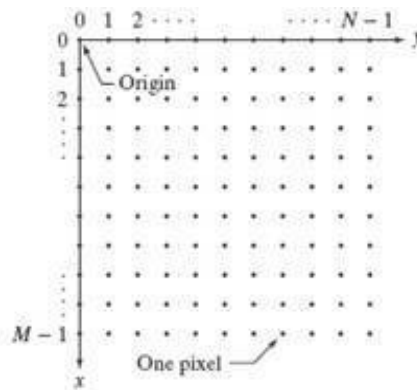
BAB II

LANDASAN TEORI

2.1 Citra Digital

Secara umum, citra digital merupakan gambar 2 dimensi yang disusun oleh data digital dalam bentuk sebuah larik (array) yang berisi nilai real maupun kompleks yang direpresentasikan dengan deretan bit tertentu. Suatu citra dapat didefinisikan sebagai fungsi $f(x,y)$ berukuran M baris dan N kolom, dengan x dan y adalah koordinat spasial, dan amplitude f di titik koordinat (x,y) dinamakan intensitas atau tingkat keabuan dari citra pada titik tersebut. (Taronisokhi, 2018)

Citra digital dibentuk oleh kumpulan titik yang dinamakan piksel (pixel atau “picture element”). Setiap piksel digambarkan sebagai satu kotak kecil. Setiap piksel mempunyai koordinat posisi. Sistem koordinat yang dipakai untuk menyatakan citra digital ditunjukkan pada Gambar 1 berikut.



Gambar 2.1. Ilustrasi Citra Digital

(*Sumber* : Taronisokhi, 2018)

Dengan sistem koordinat yang mengikuti asas pemindaian pada layar TV standar itu, sebuah piksel mempunyai koordinat berupa (x, y) dalam hal ini:

- x menyatakan posisi kolom;
- y menyatakan posisi baris;
- piksel pojok kiri-atas mempunyai koordinat $(0, 0)$ dan piksel pada pojok kanan-bawah mempunyai koordinat $(N-1, M-1)$.

Ada banyak cara untuk menyimpan citra digital di dalam memori. Cara penyimpanan menentukan jenis citra digital yang terbentuk. Format citra digital yang banyak dipakai adalah Citra Biner, Citra Grayscale, dan Citra Warna:

a. Citra Biner

Citra biner (*monochrome*) atau disebut juga *binary image*, merupakan citra digital yang setiap *pixel*-nya hanya memiliki 2 kemungkinan derajat keabuan, yaitu 0 dan 1. Nilai 0 mewakili warna hitam, dan nilai 1 mewakili warna putih, di mana setiap *pixel*-nya membutuhkan media penyimpanan sebesar 1 bit.

		0	1
		1	0

Gambar 2.2 Contoh Citra Biner Berukuran 2x2 Pixel

(Sumber : Darma Putra, 2010)

b. Citra Warna

Setiap piksel pada citra warna memiliki warna yang merupakan kombinasi dari tiga warna dasar RGB (*Red, Green, Blue*). Setiap warna dasar menggunakan penyimpanan 8 bit = 1 byte, yang berarti setiap warna mempunyai gradasi sebanyak 255 warna. Berarti setiap piksel mempunyai kombinasi warna sebanyak $28 \cdot 28 \cdot 28 = 224 = 16$ juta warna lebih. Itulah yang menjadikan alasan format ini disebut dengan *true color* karena mempunyai jumlah warna yang cukup besar sehingga bias dikatakan hampir mencakup semua warna di alam. Penyimpanan citra *true color* di dalam memori berbeda dengan citra *grayscale*. Setiap piksel dari citra *grayscale* 256 gradasi warna diwakili oleh 1 byte. Sedangkan 1 piksel citra *true color* diwakili oleh 3 byte, dimana masing-masing byte merepresentasikan warna merah, hijau dan biru.

2.2 Pengolahan Citra Digital

a. Definisi Pengolahan Citra

Pengolahan citra adalah sebuah disiplin ilmu yang mempelajari hal-hal yang berkaitan dengan perbaikan kualitas gambar (peningkatan kontras, transformasi warna, restorasi citra), transformasi gambar (rotasi, translasi, skala, transformasi geometrik), melakukan pemilihan citra ciri (*feature images*) yang optimal untuk tujuan analisis, melakukan proses penarikan informasi atau deskripsi objek atau pengenalan objek yang terkandung pada citra, melakukan kompresi atau reduksi data untuk tujuan penyimpanan data, transmisi data, dan

waktu proses data. *Input* dari pengolahan citra adalah citra, sedangkan outputnya adalah citra hasil pengolahan (Putra, Pengolahan Citra Digital, 2010).

b. Tujuan Pengolahan Citra Digital

Pengolahan citra digital banyak dimanfaatkan oleh berbagai bidang mulai dari keamanan, kesehatan, pendidikan dan bidang – bidang yang lain. Berikut beberapa tujuan dari kegiatan pengolahan citra digital.

1. Memperbaiki kualitas gambar dilihat dari aspek *radiometric* (peningkatan kontras, transformasi warna, restorasi citra) dan dari aspek *geometric* (rotasi, translasi, skala, transformasi geometrik).
2. Melakukan proses penarikan informasi atau deskripsi objek atau pengenalan objek yang terkandung pada citra.
3. Melakukan kompresi atau reduksi data untuk tujuan penyimpanan data, transmisi data, dan waktu proses data.

2.3 Metode EOF (End Of File)

Adapun algoritma dari metode EOF (End Of File) ini adalah sebagaiberikut:a. Baca informasi file, tentukan dimana posisi akhir file.b. Tandai posisi ctrl-z(penanda) sebagai awal baris penyisipan pesan.c. Sisipkan pesan dimulai dari posisi ctrl-z(penanda) hingga akhir pesan.d. Sisipkan ctrl-z(penanda) kedua pada akhir pesan.Teknik EOF tidak mengubah isi awal dari file yang disisipi. Sebagai contoh, jika kita menyisipkan sebuah pesan kedalam sebuah dokumen, isi dari document tidak akan berubah. Ini yang menjadi salah satu keunggulan metode EOF dibandingkan dengan metode steganografi yang lain.

Karena disisipkan pada akhir file, pesan yang disisipkan tidak akan bersinggungan dengan isi file, hal ini menyebabkan integrasi data dari file yang disisipi tetap terjaga. Namun, metode EOF tidak dapat menyisipkan pesan berukuran sangat besar karena dapat membuat citra berubah dan mencurigakan, baiknya pesan tidak terlalu besar agar tidak mencurigakan. Sesuai dengan konsep algoritma End of File (EoF) pada steganografi maka penanda pesan diletakkan di awal sebelum pesan disisipkan.

Metode ini menyembunyikan pesan rahasia dengan cara menambahkan bit-bit pesan yang akan disembunyikan ke akhir file citra penampung. (Ayu,2018)

.Proses penyisipan pesan dengan metode End Of File dapat dituliskan dalam algoritma sebagai berikut: (Ayu,2018)

1. Inputkan pesan yang akan disisipkan
2. Ubah pesan menjadi kode-kode desimal.
3. Inputkan citra grayscale yang akan disisipi pesan.
4. Dapatkan nilai derajat keabuan masing-masing piksel.
5. Tambahkan kode desimal pesan sebagai nilai derajat keabuan diakhir citra.
6. Petakan menjadi citra baru

2.4 UML (*Unified Modeling Language*)

UML (*Unified Modeling Language*) adalah sebuah bahasa yang sudah menjadi standar dalam industry untuk merancang, menspesifikasi dan



mendokumentasi sistem perangkat lunak. Adapun tujuan dari UML adalah: (Hendro, 2017)



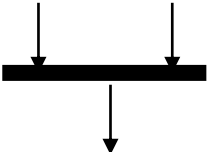
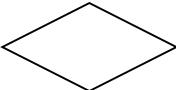
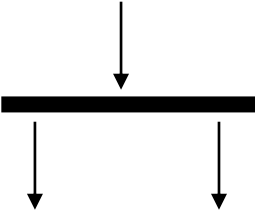




- a. Merancang perangkat lunak.
- b. Sarana komunikasi antara perangkat lunak dengan proses bisnis.
- c. Menjabarkan sistem secara rinci untuk analisa dan mencari apa yang diperlukan sistem.
- d. Mendokumentasi sistem yang ada, proses-proses dan organisasinya.

1. Activity Diagram

Activity diagram menggambarkan berbagai aliran aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, keputusan yang mungkin terjadi, dan bagaimana mereka berakhir (Hendro, 2017). Pada dasarnya, *activity* diagram merupakan variasi dari statechart diagram. *Activity* diagram mempunyai peran seperti halnya *flowchart*, akan tetapi perbedaannya dengan *flowchart* adalah *activity* diagram bisa mendukung perilaku paralel sedangkan *flowchart* tidak bisa. Berikut adalah notasi *activity* diagram.

Tabel 2.1 Notasi Activity Diagram

Simbol	Keterangan
	Titik Awal
	Titik Akhir

	<i>Activity</i>
	<i>Connector</i>
	<i>Join</i>
	Decision Pilihan untuk mengambil keputusan
	Fork; Digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu.
	Note
	Receive Signal
	Send Signal
	Option Loop

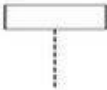


Sumber: (Hendro, 2017)

2. *Sequence Diagram*

Sequence Diagram mendeskripsikan skenario (dapat mengacu pada expanded use case yang telah dibuat) dalam bentuk diagram (Hendro, 2017). Diagram ini juga menunjukkan serangkaian pesan yang dipertukarkan oleh obyek – obyek yang melakukan suatu tugas atau aksi tertentu. Obyek – obyek tersebut kemudian diurutkan dari kiri ke kanan, aktor yang menginisiasi interaksi biasanya ditaruh di paling kiri dari diagram.

Pada diagram ini, dimensi vertikal merepresentasikan waktu. Bagian paling atas dari diagram menjadi titik awal dan waktu berjalan ke bawah sampai dengan bagian dasar dari diagram. Garis Vertical, disebut lifeline, dilekatkan pada setiap obyek atau aktor. Kemudian, lifeline tersebut digambarkan menjadi kotak ketika obyek melakukan suatu operasi, kotak tersebut disebut activation box. Obyek dikatakan mempunyai live activation pada saat tersebut. Pesan yang dipertukarkan antar obyek digambarkan sebagai sebuah anak panah antara activation box pengirim dan penerima. Kemudian di atasnya diberikan label pesan.

Tabel 2.2 Simbol Sequence Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		<i>LifeLine</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
2		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi.
3		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi.

Sumber: (Hendro, 2017)




3. Use Case Diagram







Use case diagram digunakan untuk menspesifikasikan fungsionalitas dari sistem (Eccles, 2015). *Use case* merupakan sebuah pekerjaan tertentu, misalnya login ke sistem, meng-*create* sebuah daftar belanja, dan sebagainya.


Seorang/sebuah aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu. *Use case diagram* dapat sangat membantu bila kita sedang menyusun *requirement* sebuah sistem, mengkomunikasikan rancangan dengan klien, dan merancang *test case* untuk semua *feature* yang ada pada sistem. Sebuah *use case* dapat meng-*include* fungsionalitas *use case* lain sebagai bagian dari proses dalam dirinya.

Secara umum diasumsikan bahwa *use case* yang di-*include* akan dipanggil setiap kali *use case* yang meng-*include* dieksekusi secara normal. Sebuah *use case* dapat di-*include* oleh lebih dari satu *use case* lain, sehingga duplikasi fungsionalitas dapat dihindari dengan cara menarik keluar fungsionalitas yang *common*. Sebuah *use case* juga dapat meng-*extend* *use case* lain dengan *behaviour*-nya sendiri. Sementara hubungan generalisasi antar *use case* menunjukkan bahwa *use case* yang satu merupakan spesialisasi dari yang lain.

Tabel 2.3 Simbol Use Case Diagram

Gambar	Nama	Keterangan
	<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>Use Case</i> .
	<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>Independent</i>)
	<i>Generalization</i>	Hubungan dimana objek anak (<i>Descended</i>) berbagi perilaku dan struktur data dari objek yang di atasnya

		objek induk.
	<i>Include</i>	Menspesifikasikan bahwa use case sumber secara explicit.
	<i>Extend</i>	Menspesifikasikan bahwa use case target memperluas perilaku pada use case sumber pada sebuah titik diberikan.
	<i>Assosiation</i>	Apa yang menghubungkan objek satu dengan objek yang lainnya.
	<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
	<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur dari sebuah <i>actor</i> .
	<i>Colaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya.

	<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.
---	-------------	--

Sumber: (Hendro, 2017)

2.5 Bahasa pemrograman

Bahasa pemrograman adalah perintah-perintah atau instruksi yang dimengerti oleh komputer untuk melakukan tugas tertentu. Bahasa pemrograman merupakan sebuah instruksi untuk memerintah komputer agar bisa menjalankan fungsi tertentu, namun hanya instruksi standar saja. Bahasa pemrograman juga memiliki perhimpunan dari aturan sintaks dan semantik yang tugasnya untuk mendefinisikan program komputer. Bahasa pemrograman komputer yang kita kenal antara lain adalah *Java*, *Visual Basic*, *C++*, *C*, *PHP*, dan bahasa pemrograman lainnya. Namun tentu saja kebutuhan bahasa ini harus disesuaikan dengan fungsi dan perangkat yang menggunakannya.

Menurut generasi bahasa pemrograman digolongkan menjadi 4 generasi, yaitu:

- a. Generasi ke-1: *machine language*
- b. Generasi ke-2: *assembly language: Assembler*
- c. Generasi ke-3: *high level programming language*, contoh: *C* dan *Pascal*
- d. Generasi ke-4: *4 GL (fourth-generation language)*, contoh: *SQL*
- e. Generasi ke-5: *Programming Language Based Object Oriented & Web Development*

Secara umum bahasa pemrograman dibagi menjadi 4 kelompok, yaitu :

1. *Object Oriented Language* : Seperti bahasa *Visual C, Delphi, Visual dBase, Visual FoxPro*.
2. *Low Level Language* : Bahasa *Assembly*.
3. *Middle Level Language* : Bahasa *C*.
4. *High Level Language* : Bahasa *Basic* dan *Pascal*.

Menurut tingkat kedekatannya dengan mesin komputer, bahasa pemrograman terdiri dari:

- a. Bahasa Mesin, yaitu memberikan perintah kepada komputer dengan memakai kode bahasa biner, contohnya 01100101100110.
- b. Bahasa Tingkat Rendah, atau dikenal dengan istilah bahasa rakitan (bhs.Ingggris *Assembly*), yaitu memberikan perintah kepada komputer dengan memakai kode-kode singkat (kode *mnemonic*), contohnya *MOV, SUB, CMP, JMP, JGE, JL, LOOP*, dsb.
- c. Bahasa Tingkat Menengah, yaitu bahasa komputer yang memakai campuran instruksi dalam kata-kata bahasa manusia (lihat contoh Bahasa Tingkat Tinggi di bawah) dan instruksi yang bersifat simbolik, contohnya {, }, ?, <<, >>, &&.
- d. Bahasa Tingkat Tinggi, yaitu bahasa komputer yang memakai instruksi berasal dari unsur kata-kata bahasa manusia, contohnya *begin, end, if, for, while, and, or, dsb*. Komputer dapat mengerti bahasa manusia itu diperlukan program *compiler* atau *interpreter*.

Fungsi dari bahasa pemrograman adalah untuk memerintahkan sebuah komputer agar dapat mengolah data yang sesuai dengan di inginkan. *Output* dari bahasa pemrograman ini dapat berupa aplikasi ataupun program khusus. Contoh sederhananya seperti lampu lalu lintas di jalan raya.

2.6 Visual Basic.Net

Platform Microsoft.Net merupakan model untuk *development* dimana *platform* dan aplikasi bisa dibuat dan dijalankan tanpa bergantung pada alat (*device*) yang dipakai. Teknologi ini memungkinkan beberapa aplikasi bekerja sama. *Visual Basic.Net* merupakan *core* dari pembuatan aplikasi berbasis .Net, yang merupakan lingkungan pemrograman yang mempermudah tahapan *design*, *development*, *debugging*, dan *development* dari aplikasi berbasis .Net dan *XML web service*, serta meningkatkan efisiensi *developer* dengan menyediakan lingkungan pemrograman yang sudah biasa digunakan.

Bahasa *Visual Basic* pada dasarnya adalah bahasa yang mudah dimengerti sehingga pemrograman di dalam bahasa *Basic* dapat dengan mudah dilakukan meskipun oleh orang yang baru belajar membuat program. *Visual Basic* mempunyai teknik pemrograman *visual* yang memungkinkan penggunanya untuk berkreasi lebih baik dalam menghasilkan suatu program aplikasi. Ini terlihat dari dasar pembuatan dalam *visual basic* adalah *FORM*, dimana pengguna dapat mengatur tampilan form kemudian dijalankan dalam script yang sangat mudah. Pemakaian *Visual Basic* ditandai dengan kemampuan *Visual Basic* untuk dapat berinteraksi dengan aplikasi lain di dalam sistem operasi *Windows* dengan

komponen *ActiveX Control*. Dalam *Microsoft Visual Basic.Net* terdapat dua komponen utama adalah:

1. *Net Framework Class Library*.

Komponen ini digunakan untuk menjalankan sebuah aplikasi melalui objek yang telah didefinisikan, antara lain : label, form, textbox, button, listbox, datetimer, dan lain-lain.

2. *Common Language Runtime (CLR)*

Komponen ini digunakan untuk mengeksekusi program yang ditulis dalam bahasa pemrograman yang ada dalam lingkungan *Microsoft Visual Studio.Net*, seperti: *C.Net*, *C++*, *Net*, dan juga *Visual basic.Net*.

Adapun kelebihan dan kekurangan dari *Visual Basic* ini yaitu :

1. Kelebihan *Visual Basic*

- a. *VB.Net* mempunyai fasilitas *Real Time Background Compiler* yaitu sebagai penanganan dalam error atau bug.
- b. Lebih cepat dalam pembuatan aplikasi berbasis desktop
- c. Menyediakan untuk *developer* pemrograman data akses *ActiveX Data Object (ADO)*.

2. Kelemahan *Visual Basic*

- a. Untuk versi *VB.Net 2010* dan seterusnya tidak mempunyai Komponen *Crystal Report* karena sudah terpisah.
- b. Harus ada *Net framework* agar aplikasi bisa berjalan
- c. Tidak mempunyai database sendiri.
- d. Memerlukan kapasitas yang besar untuk instalasi *VB.Net*.

2.7 *Microsoft Visual Studio*

Microsoft Visual Studio merupakan sebuah perangkat lunak lengkap (*suite*) yang dapat digunakan untuk melakukan pengembangan aplikasi, baik aplikasi bisnis, aplikasi personal, ataupun komponen aplikasi lainnya.

Visual Studio mencakup *compiler*, *SDK*, *Integrated Development Environment (IDE)*, dan dokumentasi. *Visual Studio* dapat digunakan untuk mengembangkan aplikasi dalam *native code* (dalam bentuk bahasa mesin yang berjalan di atas *windows*) ataupun *managed code* (dalam bentuk *Microsoft Intermediate Language* diatas *.NET Framework*).

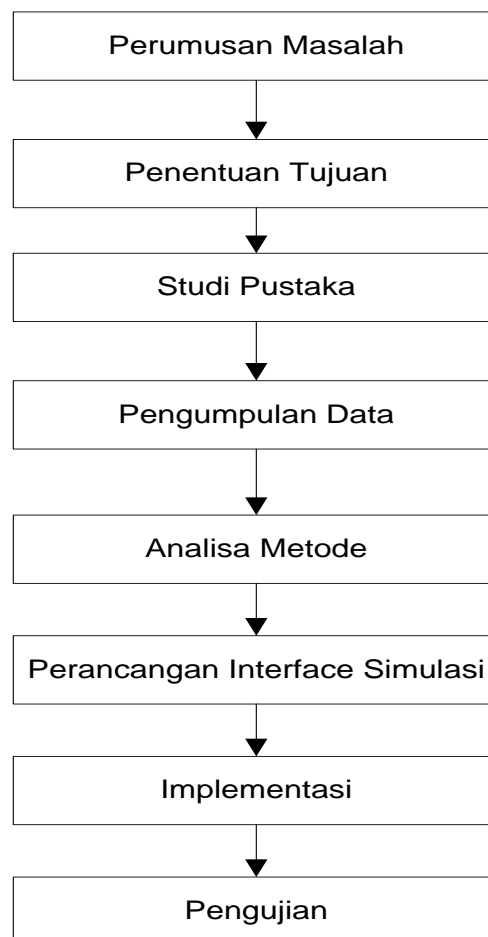
Pengguna dapat memanfaatkan *Visual Studio* untuk membuat beberapa hal seperti web dan menulis beberapa kode pemrograman seperti *Python*, *Ruby*, *C*, *C++* dan *Java*.

BAB III

METODE PENELITIAN

3.1 Tahapan Penelitian

Adapun tahapan penelitian yang dilakukan oleh penulis ini dengan judul Pengamanan File Excel Ke Dalam Gambar Dengan Algoritma LSB (Least Significant Bit) adalah sebagai berikut:



Gambar 3.1 Tahapan Penelitian

3.2 Metode Pengumpulan Data

Pengumpulan data adalah pencarian terhadap sesuatu karena ada perhatian dan keinginan terhadap hasil suatu aktivitas. Metode pengumpulan data dalam penulisan ini dibagi menjadi 2, yaitu :

1. Pengamatan (*Observation*)

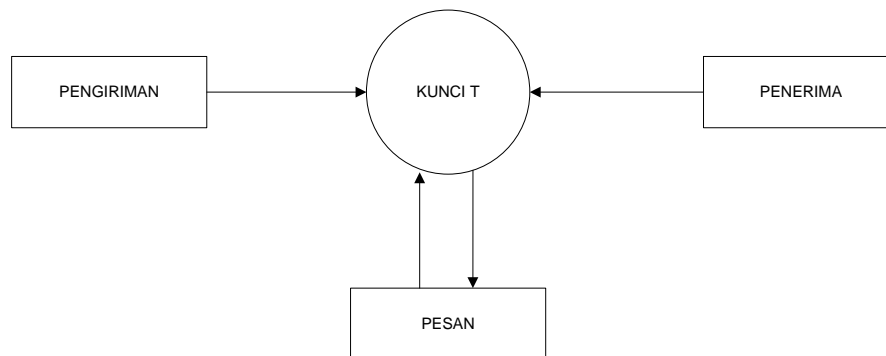
Penulis melakukan pengamatan langsung pada setiap penggunaan aplikasi chatting yang sudah ada seperti WA, BBM dan Line untuk mengamati proses keamanan yang sudah dibuat sebelumnya.

2. Penelitian Kepustakaan (*Library Research*)

Merupakan cara untuk mencari referensi dengan mengumpulkan bahan-bahan pustaka yang dilakukan di perpustakaan kampus, maupun perpustakaan umum, juga melakukan pencarian lewat internet, dengan mengunjungi situs-situs seperti *google Book online* yang dapat membantu pembahasan materi.

3.3 Analisis Permasalahan yang Berjalan

Pertukaran data dalam hal ini pesan rahasia berbentuk teks dengan menggunakan metode tradisional yaitu dengan cara bertukar kata kunci tunggal. Diagram dibawah adalah penggambaran bagaimana pertukaran pesan rahasia menggunakan kunci tunggal terjadi.



Gambar 3.2 Analisis Permasalahan yang Berjalan

Pemberitahuan kata kunci dari pengirim ke penerima menggunakan media yang umum digunakan oleh banyak orang.

3.4 Analisa Kelemahan yang Berjalan

1. Penggunaan kata kunci tunggal berpotensi terjadinya salah pemahaman. Dalam hal ini kemungkinan penerima salah mengartikan kunci yang diberikan oleh pengirim adalah hal yang dapat terjadi.
2. Pemberitahuan atau pertukaran kata kunci yang dikirimkan oleh pengirim ke penerima memiliki potensi dapat diketahui oleh orang lain sehingga pesan rahasia dapat terbongkar.

3.5 Solusi Pemecahan Masalah

Pemecahan masalah yang penulis lakukan adalah dengan melakukan penerapan metode ini yang didalamnya terdapat Algoritma *LSB*. Penggunaan metode ini dapat digunakan sebagai solusi agar pengirim dan penerima tidak lagi

harus bertukar kunci tunggal untuk membuka pesan melainkan dapat memiliki kata kunci masing-masing.

Tabel 3.1 Tabel Perencanaan Rancangan

No	Sistem yang Berjalan	Sistem yang Diusulkan	Hasil yang Ingin Dicapai
1.	Penggunaan kunci tunggal yang harus diketahui oleh pengirim dan penerima untuk membuka pesan.	Pengirim dan penerima memiliki kunci masing-masing untuk membuka pesan	Tidak ada lagi kesalahan pemahaman atau salah tafsir kunci tunggal karena pengirim dan penerima memiliki kunci yang dapat ditetapkan masing-masing pihak.
2.	Pertukaran kunci tunggal menggunakan media komunikasi yang rentan untuk dapat diketahui orang lain.	Pengirim dan penerima dapat menentukan sendiri kunci yang ingin digunakan untuk membuka pesan.	Kemungkinan bocornya kunci saat proses pertukaran informasi kunci tunggal dapat dihindari.

3.6 Analisa Proses Sistem Yang Berjalan

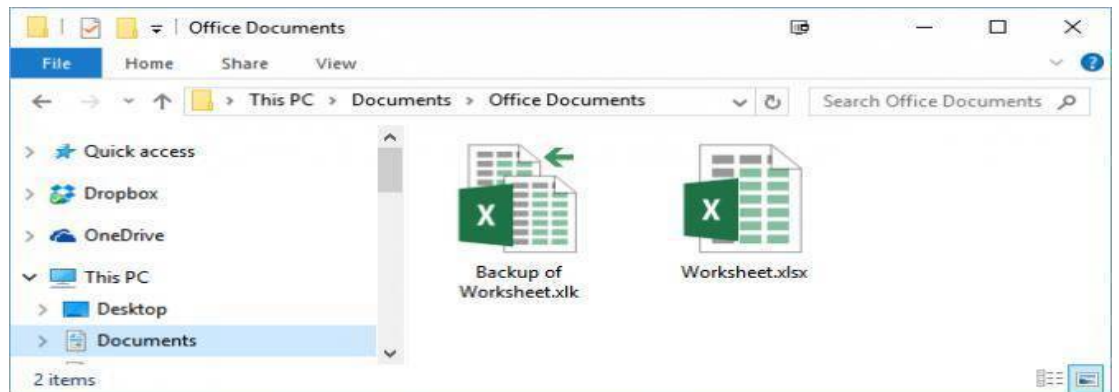
Terdapat 2 (dua) proses utama dalam penyisipan pesan menggunakan metode *Least Significant Bit*, yaitu proses *embedding* dan proses *extraction*. Proses *embedding* adalah proses penyisipan pesan rahasia ke dalam suatu media. Sedangkan proses *extraction* adalah proses pengambilan pesan rahasia dari suatu media. Pada sistem ini, pesan rahasia yang digunakan berupa *data biner* teks yang merupakan *text* dari hasil enkripsi teknik steganografi ke dalam nilai bit akhir dari media penampung (Ms. Excel *file*) dan media yang digunakan untuk penyisipan pesan adalah *file* Ms. Excel berformat .bmp, .jpg.

1. Proses *Embedding*

Proses *embedding* atau penyisipan pesan menggunakan metode *Least Significant Bit* adalah sebagai berikut :

- a) Inputkan Ms. Excel yang akan menjadi media penyisipan *text* (*cover file*).
- b) Inputkan *text* yang sudah terenkripsi untuk disisipkan.
- c) Baca nilai *biner* setiap *pixel* Ms. Excel.
- d) Sisipkan nilai *biner* dari *text* pada nilai akhir *biner* dari *pixel* Ms. Excel.
- e) Petakan menjadi Ms. Excel baru.

Berikut contoh penyisipan *text* menggunakan metode *Least Significant Bit*: Terdapat satu pesan yang sudah dienkripsi “AKU” yang akan disisipkan pada suatu Ms. Excel.



Gambar 3.3 worksheet.xlsx

Langkah pertama adalah mengubah kedua data tersebut (kata AKU dan Ms. Excel) menjadi biner.

Tabel 3.2

Nilai Biner AKU		
A	K	U
0	0	0
1	1	1
0	0	0
0	0	1
0	1	0
0	0	1
0	1	0
1	1	1

Tabel 3.3 Tabel Biner Ms. Excel

000000 01	000101 00	000000 00	000000 01	000101 00	000000 00	000000 01	000101 00
000000 01	000000 00	000100 11	000000 00	000000 00	000100 11	000000 00	000000 00
000101 01	000000 00	000000 00	000101 10	000000 01	000000 00	000110 00	000000 00
000000 00	000110 10	000000 00	000000 01	000101 00	000000 00	000000 00	000100 11
000000 00	000000 00	000100 11	000000 00	000000 00	000101 10	000000 01	000000 00
000101 10	000000 01	000000 00	000101 10	000000 01	000000 10	000101 01	000000 10
000000 00	000100 11	000000 00	000000 01	000100 11	000000 11	000000 00	000100 01

000000 01	000000 00	000100 01	000000 01	000000 00	000100 00	000000 00	000000 00
--------------	--------------	--------------	--------------	--------------	--------------	--------------	--------------

Kemudian gantikan tiap biner dari teks nya ke dalam akhir biner Ms. Excel penampung, sehingga akan terlihat seperti pada tabel berikut ini.

Tabel 3.4 Tabel biner Ms. Excel yang berisi pesan rahasia

000000 00	000000 01	000100 10	000000 00	000000 00	000100 10	000000 00	0000 0001	A
000101 00	000000 01	000000 00	000101 10	000000 01	000000 00	000110 01	0000 0001	K
000000 00	000110 11	000000 00	000000 01	000110 00	000000 01	000000 00	0001 1001	U
000000 00	000000 00	000101 01	000000 00	000000 00	000100 11	000000 00	0000 0000	-
000100 11	000000 00	000000 00	000101 11	000000 01	000000 00	000101 11	0000 0001	-
000000 00	000101 11	000000 01	000000 10	000101 01	000000 10	000000 00	0001 0011	-
000000 00	000000 00	000100 11	000000 11	000000 00	000100 01	000000 01	0000 0000	-
000100 01	000000 01	000000 00	000100 01	000000 00	000000 00	000100 01	0000 0000	-

Terlihat pada tiap akhir dari biner Ms. Excel telah tersisipi oleh pesan rahasia yang ditandai dengan huruf *Bold* (cetak tebal). Langkah selanjutnya adalah matriks tersebut akan dipetakan kembali dalam bentuk Ms. Excel dan Ms. Excel ini disebut *stego file*.

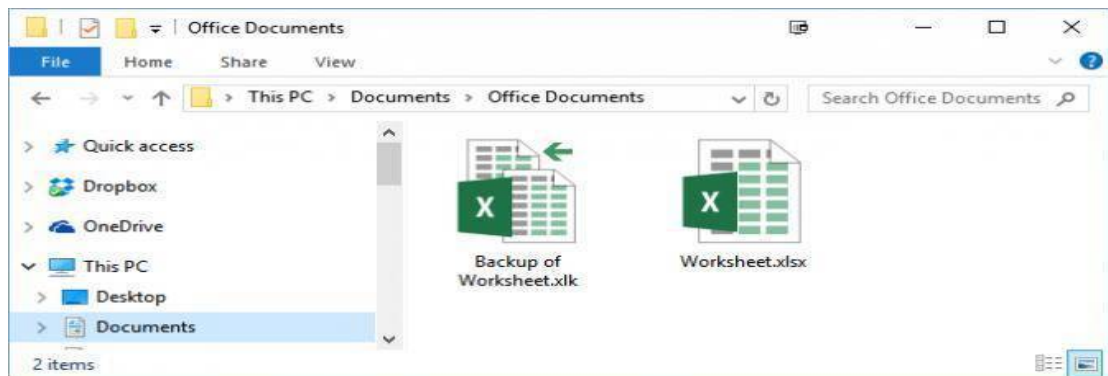
2. Proses *Extraction*

Proses *extraction* atau pengambilan *text* dari media penampung menggunakan metode *Least Significant Bit* adalah sebagai berikut :

1. Masukkan Ms. Excel yang telah disisipkan *text* (*stego file*).
2. Baca nilai biner dari pixel *stego file* yang terdapat pada biner terakhir *pixel* Ms. Excel penampung.

3. Ambil nilai *binertext* yang terdapat pada *stego file*, yaitu nilai *biner* dari tiap-tiap pixel terakhir yang berubah.

Berikut contoh pengambilan *text* dengan menggunakan metode *Least Significant Bit*: Terdapat suatu Ms. Excel “contoh.bmp” yang telah disisipkan *text* (*stego file*). Nilai setiap *pixel file* Ms. Excel tersebut dapat dilihat pada Tabel 7.



Gambar 3.4 File Excel

Kemudian *text* dibaca pada nilai akhir dari *biner pixel stego file* seperti pada tabel 3.5

Tabel 3.5 Tabel biner Ms. Excel yang berisi pesan rahasia

000000 00	000000 01	000100 10	000000 00	000000 00	000100 10	000000 00	000000 01	A
000101 00	000000 01	000000 00	000101 10	000000 01	000000 00	000110 01	000000 01	K
000000 00	000110 11	000000 00	000000 01	000110 00	000000 01	000000 00	000110 01	U
000000 00	000000 00	000101 01	000000 00	000000 00	000100 11	000000 00	000000 00	-
000100 11	000000 00	000000 00	000101 11	000000 01	000000 00	000101 11	000000 01	-
000000 00	000101 11	000000 01	000000 10	000101 01	000000 10	000000 00	000100 11	-
000000 00	000000 00	000100 11	000000 11	000000 00	000100 01	000000 01	000000 00	-

000100 01	000000 01	000000 00	000100 01	000000 00	000000 00	000100 01	000000 00	-
--------------	--------------	--------------	--------------	--------------	--------------	--------------	--------------	---

Dengan mengambil nilai biner pixel yang terakhir,yang dimulai dari awal pada baris pertama pixel Ms. Excel, didapatkan nilai biner dari text yaitu “01000001=A, 01001011=K, 01010101=U”

Tabel 3.6 Tabel biner pesan rahasia yang disisipkan

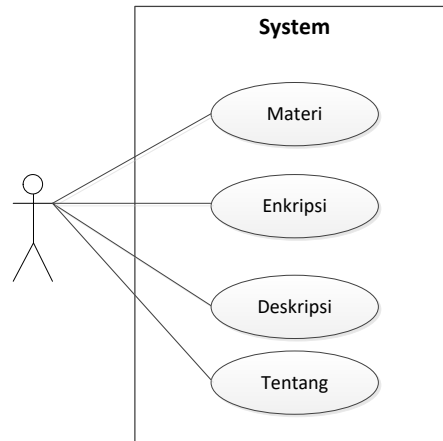
000000 00	000000 01	000100 10	000000 00	000000 00	000100 10	000000 00	000000 01	A
000101 00	000000 01	000000 00	000101 10	000000 01	000000 00	000110 01	000000 01	K
000000 00	000110 11	000000 00	000000 01	000110 00	000000 01	000000 00	000110 01	U

3.7 Perancangan Berorientasi Obejk

Perancangan atau Pemodelan Berorientasi Ojek merupakan proses mendapatkan informasi dari model dan menampilkannya secara grafik dengan menggunakan sebuah standar elemen grafik. Tujuan dari perancangan berorientasi ojbek ini memungkinkan adanya komunikasi yang lebih berkualitas antara pengguna, pengembang penganalisis, tetster, manajer dan siapapun yang terlibat dalam proyek pengembangan sistem informasi.

1. Use case Diagram

Berikut adalah use case diagram yang menggambarkan kegiatan.



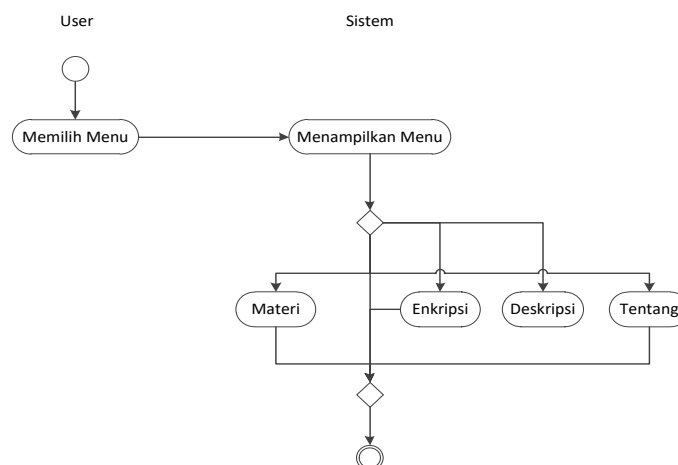
Gambar 3.5. Use Case Diagram

Keterangan :

Dalam use case diagram di atas, user/pengguna sebagai actor yang mempunyai use case Materi, Enkripsi dan Tentang.

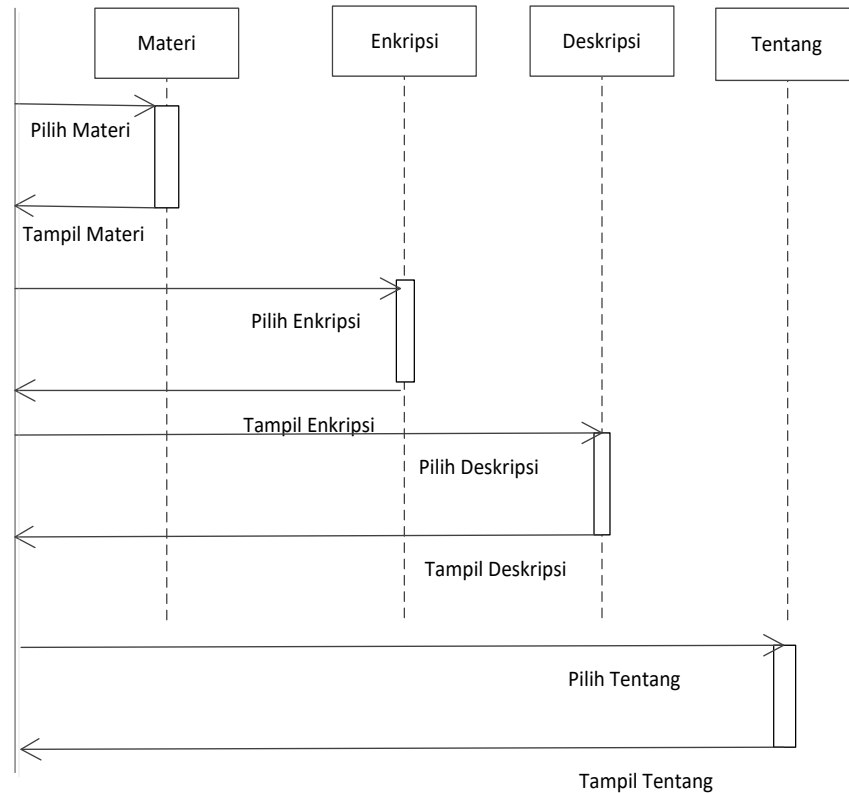
2. Activity Diagram

Activity diagram menggambarkan aktifitas-aktifitas yang terjadi dalam aplikasi dari aktivitas dimulai sampai aktivitas berhenti.



Gambar 3.6 Activity Diagram

3. Sequence Diagram



Gambar 3.7 *Sequence Diagram*

Keterangan Gambar :

1. Dalam diagram di atas menjelaskan bahwa user memilih materi kemudian Sistem menampilkan materi yang berkaitan dengan materi
2. User merequest Enkripsi kemudian Sistem menampilkan menu Enkripsi
3. User merequest Deskripsi kemudian Sistem menampilkan menu Deskripsi
4. User merequest Menu Tentang kemudian Sistem menampilkan Form Tentang.

3.8 Perancangan Antarmuka

Perancangan antarmuka menampilkan antarmuka *sistem* yang akan digunakan oleh pengguna. Tahapan ini berguna untuk memudahkan pengguna dalam menggunakan sistem ini. Berikut rancangan antarmuka sistem keamanan yang akan dibangun.

3.8.1 Halaman Steganografi dengan LSB

Halaman menu utama merupakan halaman yang pertama kali muncul saat sistem ini dijalankan. Halaman ini memiliki 4 (empat) buah menu bar, yaitu menu *File*, *About*, *Help* dan *Exit*. Pada halaman menu bar *File* terdapat 2 buah submenu, submenu *Embed/Extract* dan submenu *Close*. Tampilan rancangan form Menu utama dapat dilihat pada Gambar 3.4.

The screenshot shows a graphical user interface for an LSB Steganography application. The interface is organized into several sections:

- File Input:** Two rows for file selection. The first row is labeled "File Gambar" and the second "File Steganografi". Each row contains an "INPUT GAMBAR" text box and a "Buka" button.
- Password and Message Length:** A "Password (Key) :" label with a "KEY" text box. Below it is a "Panjang Pesan" label with a text box containing "10000".
- Image Display:** Two large rectangular boxes labeled "GAMBAR ASLI" and "GAMBAR STEGANOGRAFI".
- Message and Processing:** A "Pesan :" label with a text box. To its right is a "PROSES" button. Further right are labels for "Maksimum Karakter Sisa", "Jumlah Karakter", and "Jumlah Bit".
- Action Buttons:** A "BUKA GAMBAR Steganografi" button, an "AMBIL PESAN" button, and a "KELUAR" button.

Gambar 3.8 Rancangan Form Menu Utama

BAB IV

HASIL DAN PEMBAHASAN

4.1 Implementasi Sistem

4.1.1 Analisa Kebutuhan Sistem

Analisis kebutuhan sistem merupakan analisis yang dibutuhkan untuk menentukan spesifikasi kebutuhan sistem. Spesifikasi ini juga meliputi elemen atau komponen – komponen apa saja yang dibutuhkan untuk sistem yang akan dibangun sampai dengan sistem tersebut diimplementasikan. Analisis kebutuhan ini juga menentukan spesifikasi masukan yang diperlukan sistem, keluaran yang akan dihasilkan sistem dan proses yang dibutuhkan untuk mengolah masukan sehingga menghasilkan suatu keluaran yang diinginkan.

1. Analisis Perangkat Keras (Hardware)

Perangkat keras minimum yang digunakan untuk membangun Sistem Informasi Penjualan ini adalah

- a. Processor berkecepatan 2.0 Ghz
- b. RAM 2 Gb
- c. Hardisk minimal 10 Gb untuk menyimpan data
- d. LAN Card
- e. Keyboard dan Mouse
- f. Monitor 14.

2. Analisis Perangkat Lunak (Software)

Untuk mendukung dalam penyimpanan informasi, dibutuhkan suatu fasilitas yang memadai. Yaitu berupa perangkat lunak (software) yang dirancang untuk memudahkan dalam pembangunan dan menjalankan sisten nantinya. Adapun perangkat lunak yang digunakan adalah sebagai berikut :

- a. Microsoft Windows 7 , Windows XP sebagai sistem operasi
- b. Microsoft Visual Studio 2010

4.1.2 Algoritma *Least Significant Bit* (LSB)

Algoritma least significant bit adalah algoritma yang di gunakan untuk menyisipkan data atau mengambil data dari dalam media penyimpanan yang digunakan. Algoritma steganografi *LSB* dibagi menjadi dua, yaitu menyisipkan data teks (*Embedded*) dan mengabil data teks (*Extraction*).

1. Proses Penyisipan Data Teks (*Embedded*)

Algoritma atau langkah-langkah untuk menyisipkan data teks pada data citra digital:

Input : C, T, KD, Pc, Pb, vM, vH, vB, toLSB, toDesimal, toBiner, xpix,

Gp

Output : CT

Proses :

for Pc = 0 To panjang C -1

for Pb = 0 To panjang C -1

vM = C. Gp (Pb dan Pc) R

$$vH = C. Gp (Pb \text{ dan } Pc) G$$

$$vB = C. Gp (Pb \text{ dan } Pc) B$$

$$T1 = \text{Mid } i, 1$$

$$T2 = \text{Mid } i + 1, 1$$

$$T3 = \text{Mid } i + 2, 1$$

$$vM = \text{toDecimal}(\text{toLSB}(\text{ToBiner}(vM), T1))$$

$$vH = \text{toDecimal}(\text{toLSB}(\text{ToBiner}(vH), T2))$$

$$vB = \text{toDecimal}(\text{toLSB}(\text{ToBiner}(vB), T3))$$

$$xpix = xpix + 1$$

If $xpix > xpx$ Then Exit For

$$i = i + 3$$

Next

$$CT \leftarrow \text{Stego Image (gambar yang telah berisi pesan)}$$

2. Proses Mengambil Data Teks (*Extraction*)

Algoritma atau langkah-langkah untuk membaca pesan pada data citra digital adalah sebagai berikut:

Input : SI, T, Pc, Pb, vM, vH, vB, toBiner, xpix, Gp, Gpes

Output : EP

Proses :

$$\text{For } Pc = 0 \text{ To } SI.\text{Height} - 1$$

$$\text{For } Pb = 0 \text{ To } SI.\text{Height} - 1$$

$$vM = SI.Gp(Pb, Pc).R$$

$$vH = SI.GP(Pb, Pc).G$$

$$vB = SI.GP(Pb, Pc).B$$

$$T = T.Mid((ToBiner(vM)), 8, 1)$$

$$T.Mid((ToBiner(vH)), 8, 1)$$

$$T.Mid((ToBiner(vB)), 8, 1)$$

$$xpix = xpix + 1$$

If $xpix > xpx$ Then Exit For

Next

$$T = T + 1 * 8$$

Next

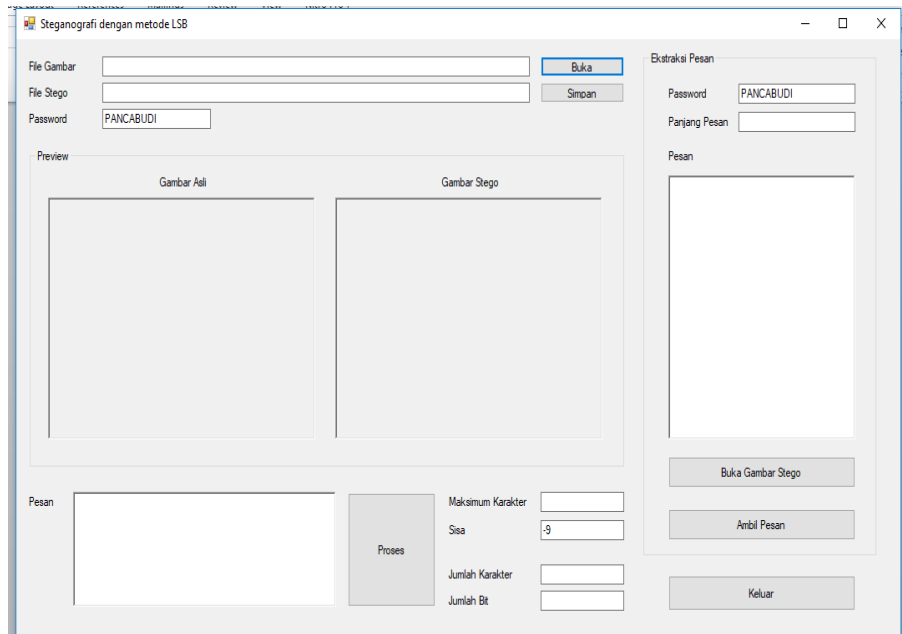
EP ← Pesan teks yang di ekstrak

4.2 Implementasi Sistem

Tahap implementasi merupakan lanjutan dari tahap perancangan sistem. Pada tahap ini dilakukan implementasi sistem ke dalam bahasa pemrograman berdasarkan hasil analisa dan perancangan sistem. Pada tahap implementasi ini digunakan perangkat lunak dan perangkat keras, sehingga sistem yang dibangun dapat diselesaikan dengan baik.

4.2.1 Tampilan Halaman Steganografi

Halaman Steganografi merupakan halaman yang muncul pertama sekali pada saat sistem dijalankan. Tampilan halaman Steganografi dapat dilihat pada Gambar 4.1.



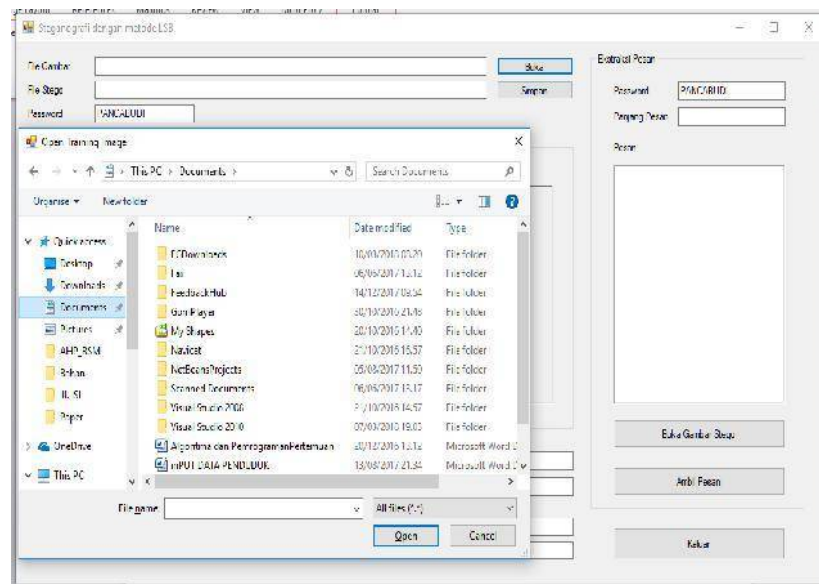
Gambar 4.1. Tampilan Halaman Menu Utama

Keterangan:

1. *Buka* untuk mencari file gambar dengan format jpg, bmp, png.
2. *Simpan* untuk menyimpan gambar yang telah disisipkan pesan text.
3. *Picture Box* (Gambar Asli) untuk menampilkan gambar asli sebelum disisipkan pesan text.
4. *Picture Box* (Gambar Stego) untuk menampilkan gambar asli sesudah disisipkan pesan text.
5. *Text Box* (Pesan) untuk menginputkan pesan yang akan disisipkan ke file gambar.
6. *Button* (Proses) untuk memproses penyisipan pesan ke dalam gambar dengan metode Steganografi.

4.2.2 Tampilan Cari Gambar

Halaman Cari Gambar merupakan halaman yang muncul pada saat proses untuk menyembunyikan pesan. Tampilan halaman Cari Gambar dapat dilihat pada Gambar 4.2.



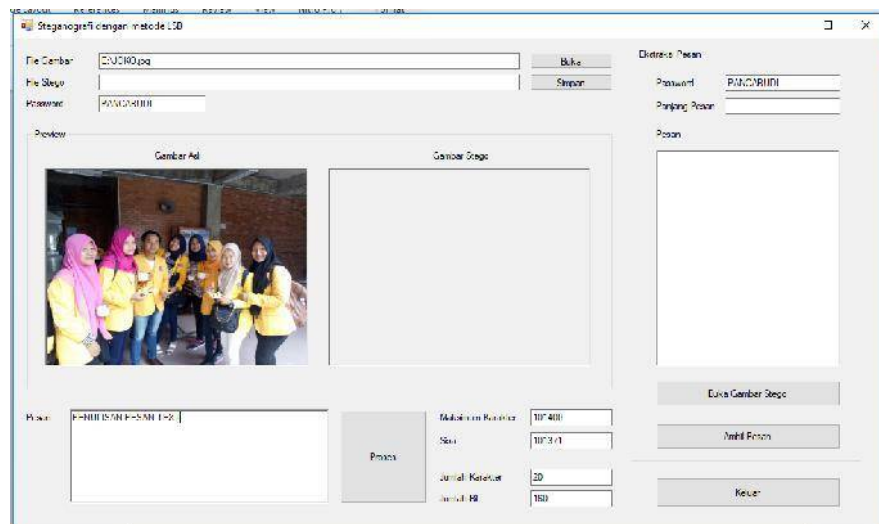
Gambar 4.2 Tampilan Cari Gambar

Keterangan:

1. Klik Buka pada *Text Box* File Gambar
2. Lalu, pilih gambar yang akan disisipkan oleh pesan text, dengan file gambar bertipe JPGE.
3. Lalu Klik OK.

4.2.3 Tampilan Penyembunyian Pesan Text

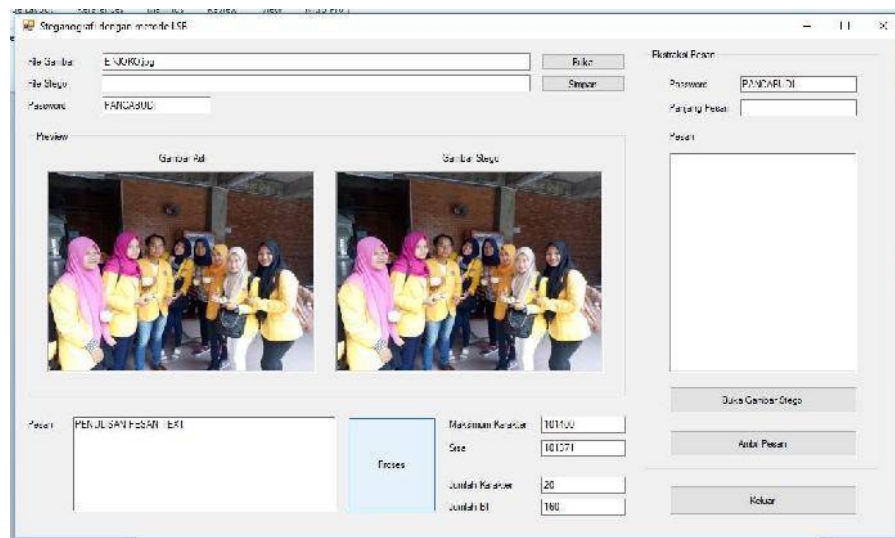
Halaman Penyembunyian Pesan Text merupakan halaman yang muncul pada saat proses untuk penyembunyian pesan. Tampilan halaman Penyembunyian Pesan Text dapat dilihat pada Gambar 4.3.



Gambar 4.3. Tampilan Penyembunyian Pesan Text

Keterangan:

1. Klik Buka pada *Text Box* File Gambar
2. Lalu, pilih gambar yang akan disisipkan oleh pesan text, dengan file gambar bertipe JPGE.
3. Lalu Klik OK.
4. Setelah muncul file gambar pada *Picture Box* (Gambar Asli), maka ketikkan pesan pada *Text Box* Pesan.
5. Lalu Proses.
6. File Gambar pada *Picture Box* (Gambar Asli) akan membuat file gambar baru yang muncul pada *Picture Box* (Gambar Stego).



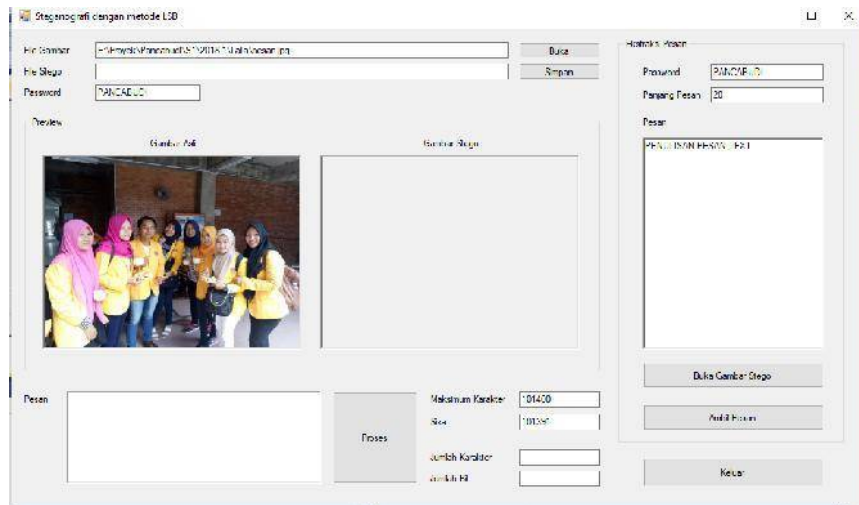
Gambar 4.4. Tampilan Gambar Yang Tersimpan Text

Keterangan:

1. Klik Buka pada *Button* “Buka Gambar Stego”
2. Lalu pilih gambar untuk melihat pesan.
3. Setelah gambar muncul pada *Picture Box* (Gambar Asli), maka masukkan password sesuai dengan pada saat proses penyisipan pesan.
4. Setelah password di inputkan, maka klik *Button* “Ambil Pesan”.
5. Maka pesan yang tersimpan pada gambar, akan muncul pada *Text Box* pesan.
6. Selesai.

4.2.4 Tampilan Menampilkan Pesan Text

Halaman Menampilkan Pesan Text merupakan halaman yang muncul pada saat proses untuk menampilkan pesan. Tampilan halaman menampilkan Pesan Text dapat dilihat pada Gambar 4.5.



Gambar 4.5. Tampilan Penyembunyian Pesan Text

Keterangan:

1. Klik Buka pada *Button* File Buka Gambar Stego
2. Lalu, pilih gambar yang telah disisipkan oleh pesan text, dengan file gambar bertipe JPGE.
3. Lalu Klik OK.
4. Setelah muncul file gambar pada *Picture Box* (Gambar Asli), maka klik button ‘Ambil Pesan’.
5. Lalu Proses.
6. Maka pesan yang tadinya tersisipkan pada gambar, muncul pada *textbox* pesan.

4.3 Pengujian Sistem

Perangkat lunak adalah elemen kritis dari jaminan kualitas perangkat lunak dan merepresentasikan kajian pokok dari spesifikasi, perancangan, dan pengkodean. Pengujian yang digunakan untuk menguji sistem ini adalah metode pengujian *black-box*. Pengujian *black-box* berfokus pada persyaratan fungsional perangkat lunak.

4.3.1 Rencana Pengujian

Pengujian fungsi Implementasi Steganografi Lsb Pada Penyembunyian Pesan Teks Pada Citra Digital ini dilakukan dengan menggunakan metode Black Box. Pengujian dilakukan pada fungsi-fungsi sistem untuk menentukan apakah fungsi tersebut telah berjalan sesuai dengan yang diharapkan.

1) Rencana Pengujian Cari Gambar

Tabel 4.1 Rencana Pengujian Cari Gambar

Menu yang diuji	Detail pengujian	Jenis uji
Menu Utama	Tampilan Halaman Awal	<i>Black box</i>
Mengelola proses penyembunyian pesan text	Input Gambar	<i>Black box</i>
	Input Pesan	<i>Black box</i>
	Input Password	<i>Black box</i>

2) Rencana Pengujian Pengujian Pengguna

Tabel 4.2 Rencana Pengujian Pengguna (*User*)

Menu yang diuji	Detail pengujian	Jenis uji
Input Password	Menginputkan Key Pada Pesan	<i>Black box</i>
Input Gambar	Mencari Gambar Untuk Media Pesan	<i>Black box</i>
Input Pesan	Menampilkan Pesan yang ada pada Gambar.	<i>Black box</i>

4.3.2 Rencana Pengujian

Rencana pengujian yang telah disusun, maka dapat dilakukan pengujian sebagai berikut :

1) Input Gambar

Tombol cari gambar diuji untuk melihat efektifitas dari button tersebut, apakah button berfungsi dengan baik. Hasil uji dapat dilihat pada tabel berikut :

Table 4.3. Pengujian Input Gambar

Nama fungsi	Buka (File Gambar)
Tujuan	Untuk menguji link berfungsi dengan baik
Aktor	Pengguna (<i>user</i>)
Kondisi awal	Berada dihalaman utama
Kondisi akhir	File Gambar Muncul Pada <i>Picture Box</i>
Skenario	<ol style="list-style-type: none"> 1) Aktor menekan Button Buka, dengan Text Box File Gambar 2) Sistem akan memunculkan Tampilan Explore Windows untuk mencari gambar yang ada pada PC atau Komputer 3) Jika sudah menemukan gambar, klik OK. Maka gambar akan masuk kedalam sistem.
Hasil yang didapat	Gambar Muncul pada Sistem
Kesimpulan	Fungsi berjalan dengan baik

2) Input Pesan

Tombol input pesan diuji untuk melihat efektifitas dari button tersebut, apakah button berfungsi dengan baik. Hasil uji dapat dilihat pada tabel berikut :

Tabel 4.4. Pengujian Input Pesan

Nama fungsi	Proses (Button)
Tujuan	Untuk menguji apakah proses tersebut sesuai dengan yang diinginkan
Aktor	Pengguna (<i>user</i>)
Kondisi awal	Berada pada Menu Utama
Kondisi akhir	Menghasilkan Pesan pada gambar yang sudah tersisipkan text.
Skenario	<ol style="list-style-type: none"> 1) Aktor menginputkan pesan text pada text box proses 2) Sistem akan menyisipkan pesan tersebut kedalam gambar, dan akan menampilkan gambar tersebut di Picture Box Steganografi. 3) Lalu, klik simpan untuk menyimpan gambar yang telah di sisipkan text.
Hasil yang didapat	Gambar yang telah disisipkan Pesan (Button Simpan)
Kesimpulan	Fungsi berjalan dengan baik

3) Input Password

Tombol input password diuji untuk melihat efektifitas dari textbox tersebut, apakah textbox berfungsi dengan baik. Hasil uji dapat dilihat pada tabel berikut :

Tabel 4.5 Pengujian Input Password

Nama fungsi	Text Box (Password)
Tujuan	Untuk menguji apakah proses tersebut sesuai dengan yang diinginkan
Aktor	Pengguna (<i>user</i>)
Kondisi awal	Berada pada Menu Utama
Kondisi akhir	Menghasilkan Password pada gambar yang sudah tersisipkan text untuk keamanan pesan.
Skenario	<ol style="list-style-type: none"> 1. Aktor menginputkan Password pada text box Password 2. Sistem akan memberikan keamanan tersebut kedalam gambar, dan meminta konfirmasi password saat akan menampilkan gambar tersebut di Picture Box Steganografi.
Hasil yang didapat	Password pada gambar (Button Simpan)
Kesimpulan	Fungsi berjalan dengan baik

4) Menampilkan Pesan

Menampilkan Pesan diuji untuk melihat efektifitas dari textbox tersebut, apakah textbox berfungsi dengan baik. Hasil uji dapat dilihat pada tabel berikut :

Tabel 4.6. Pengujian Menampilkan Pesan

Nama fungsi	Buka Gambar Stegano
Tujuan	Untuk menguji apakah proses tersebut sesuai dengan yang diinginkan
Aktor	Pengguna (<i>user</i>)
Kondisi awal	Berada pada Menu Utama
Kondisi akhir	Menghasilkan pesan text yang dihasilkan dari gambar stego.
Skenario	<ol style="list-style-type: none"> 1. Aktor mengklik button ‘Ambil Gambar Stego’, 2. Lalu, masukkan password pada textbox password. 3. Setelah itu, klik button ‘Ambil Pesan’, jika sesuai password dengan gambar, maka pesan akan muncul pada textbox pesan.

Hasil yang didapat	Pesan text pada text box stego.
Kesimpulan	Fungsi berjalan dengan baik

4.3.4 Kesimpulan dan hasil pengujian alpha

Hasil pengujian dari pengujian sistem telah selesai, menunjukkan bahwa sistem sudah memenuhi syarat fungsional. Secara fungsional sistem yang sudah dibangun sudah dapat menghasilkan keluaran sesuai yang diharapkan.

Tabel 4.7. Kesimpulan Pengujian Sistem

Nama fungsi	Hasil
Password	Fungsi berjalan dengan baik
Menampilkanl Pesan	Fungsi berjalan dengan baik
Input Gambar	Fungsi berjalan dengan baik
Input Pesan	Fungsi berjalan dengan baik
Input Password	Fungsi berjalan dengan baik

BAB V

PENUTUP

1. Kesimpulan

Setelah keseluruhan proses dilakukan, yaitu dimulai dari tahapan studi literatur hingga pengujian perangkat lunak, maka dapat diambil kesimpulan sebagai berikut:

1. Penerapan metode EOF dengan menggunakan citra dalam proses penyembunyian pesan sangat berfungsi dengan baik, dikarenakan pesan yang disembunyikan pada citra digital sulit dilihat dan dideteksi.
2. Proses kinerja penyembunyian pesan dengan metode EOF menggunakan biner dari citra yaitu nilai dari Red, Green, dan Blue (RGB), dengan nilai biner dari RGB tersebut pesan dapat disisipkan dengan menggunakan konversi nilai karakter dirubah menjadi biner menggunakan tabel ASCII.

2. Saran

Adapun saran-saran yang dapat penulis berikan untuk pengembangan dan perbaikan sistem ini adalah sebagai berikut :

1. Penelitian ini dapat dikembangkan dengan mencoba menerapkan beberapa metode lainnya seperti (algoritma RSA dan DES) sehingga pendeteksian pesan tersembunyi pada sebuah gambar lebih akurat dan sulit untuk dipecahkan.
2. Pada proses penyembunyian pesan sebaiknya dikombinasi dengan metode lainnya agar pesan yang disisipkan pada gambar menjadi lebih aman.

DAFTAR PUSTAKA

- Arifin, M., & HS, R. H. H. (2017). Perancangan Sistem Informasi Pusat Karir Sebagai Upaya Meningkatkan Relevansi Antara Lulusan Dengan Dunia Kerja Menggunakan UML. *IC-Tech*, 12(2).
- Gunawan, I., Sumarno, S., Irawan, E., & Tambunan, H. S. (2018). Pengamanan Berkas Dokumen Menggunakan Fungsi Algoritma Steganografi LSB. *Algoritma: Jurnal Ilmu Komputer Dan Informatika*, 2(1).
- Hartanto, S. (2017). Implementasi fuzzy rule based system untuk klasifikasi buah mangga. *TECHSI-Jurnal Teknik Informatika*, 9(2), 103-122.
- Harumy, T. H. F., & Sulistianingsih, I. (2016). Sistem penunjang keputusan penentuan jabatan manager menggunakan metode mfep pada cv. Sapo durin. In *Seminar Nasional Teknologi Informasi dan Multimedia* (pp. 6-7).
- Havena, M., & Marlina, L. (2018). The Technology of Corn Processing as an Effort to Increase The Income of Kelambir V Village. *Journal of Saintech Transfer*, 1(1), 27-32.
- Indra permana, A. M. I. N. U. D. D. I. N. "Sistem pakar mendeteksi hama dan penyakit tanaman kelapa sawit pada pt. moeis kebun sipare-pare kabupaten batubara." (2013).
- Khairul, K., IlhamiArsyah, U., Wijaya, R. F., & Utomo, R. B. (2018, September). Implementasi augmented reality sebagai media promosi penjualan rumaH. In *Seminar Nasional Royal (SENAR)* (Vol. 1, No. 1, pp. 429-434).
- Kurnia, D. (2017). Analisis QoS Pada Pembagian Bandwidth Dengan Metode Layer 7 Protocol, PCQ, HTB Dan Hotspot Di SMK Swasta Al-Washliyah Pasar Senen. *CESS (Journal of Computer Engineering, System and Science)*, 2(2), 102-111.
- Kurniawan, H. (2018). Pengenalan Struktur Baru untuk Web Mining dan Personalisasi Halaman Web. *Jurnal Teknik dan Informatika*, 5(2), 13-19.
- Mariance, U. C. (2018). Analisa dan Perancangan Media Promosi dan Pemasaran Berbasis Web Menggunakan Work System Framework (Studi Kasus di Toko Mandiri Prabot Kota Medan). *Jurnal Ilmiah Core IT: Community Research Information Technology*, 6(1).

- Marlina, L., Muslim, M., Siahaan, A. U., & Utama, P. (2016). Data Mining Classification Comparison (Naïve Bayes and C4. 5 Algorithms). *Int. J. Eng. Trends Technol*, 38(7), 380-383.
- Mayasari, Nova. "Comparison of Support Vector Machine and Decision Tree in Predicting On-Time Graduation (Case Study: Universitas Pembangunan Panca Budi)." *Int. J. Recent Trends Eng. Res* 2.12 (2016): 140-151.
- Muttaqin, Muhammad. "Analisa pemanfaatan sistem informasi e-office pada universitas pembangunan panca budi medan dengan menggunakan metode utaut." *Jurnal Teknik dan Informatika* 5.1 (2018): 40-43.
- Putra, D. (2010). *Pengolahan citra digital*. Penerbit Andi.
- Perwitasari, I. D. (2018). Teknik Marker Based Tracking Augmented Reality untuk Visualisasi Anatomi Organ Tubuh Manusia Berbasis Android. *INTECOMS: Journal of Information Technology and Computer Science*, 1(1), 8-18.
- Puspita, Khairani, and Purwa Hasan Putra. "Penerapan Metode Simple Additive Weighting (SAW) Dalam Menentukan Pendirian Lokasi Gramedia Di Sumatera Utara." *Seminar Nasional Teknologi Informasi Dan Multimedia*, ISSN. 2015.
- Putera, A., Siahaan, U., & Rahim, R. (2016). Dynamic key matrix of hill cipher using genetic algorithm. *Int. J. Secur. Its Appl*, 10(8), 173-180.
- Putra, Randi Rian, and Cendra Wadisman. "Implementasi Data Mining Pemilihan Pelanggan Potensial Menggunakan Algoritma K Means." *INTECOMS: Journal of Information Technology and Computer Science* 1.1 (2018): 72-77.
- Wahyudi, H. (2013). Perancangan Sistem Informasi Penyerangan Siswa Baru SMU Menggunakan PHP Dan Mysql. *Jurnal Computech & Bisnis*, 7(2), 84-95.
- Widodo, A. P., Sutanto, T., & Subriadi, A. P. (2016). A Model To Determine Cost Estimation For Software Development Projects Of Small And Medium Scales Using Use Case Points. *Journal of Theoretical & Applied Information Technology*, 85(1).
- Wiliani, N. (2018). Rancang Bangun Aplikasi Kasir Tiket Nonton Bola Bareng Pada X Kasir Di Suatu Lokasi X Dengan Visual Basic 2010 Dan Mysql. *Rekayasa Informasi*, 6(2).
- Zebua, T. (2018). Penerapan Metode LSB-2 untuk Menyembunyikan Ciphertext Pada Citra Digital. *Pelita Informatika: Informasi dan Informatika*, 10(3).