ANALISIS TEKNIK STEGANOGRAFI PADA AUDIO MP3 MENGGUNAKAN METODE PARITY CODING DAN ENKRIPSI VIGENERE CIPHER

SKRIPSI

Diajukan Untuk Memenuhi Salah Satu Syarat Untuk Memperoleh
Gelar Sarjana Teknik Pada Fakultas Teknik
Universitas Islam Riau

OLEH :

M FADLI

143510524

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS ISLAM RIAU
PEKANBARU
2020

KATA PENGANTAR

Assalaamu'alaikum Wr.Wb.

Segala puji bagi Allah SWT yang selalu memberikan rahmat dan hidayah-Nya serta nikmat yang tak terhingga, sehingga penulis dapat menyelesaikan proposal ini dengan judul "Analisis Teknik Steganografi Pada Audio MP3 Menggunakan Metode *Parity Coding* dan Enkripsi *Vigenere Cipher*" untuk memenuhi salah satu syarat menyelesaikan studi serta dalam rangka memperoleh gelar Sarjana Pendidikan Strata Satu pada Program Studi Teknik Informatika Fakultas Teknik Provinsi Riau.

Dalam penyusunan laporan skripsi ini, penulis sadar bahwa tanpa bantuan dan bimbingan berbagai pihak maka laporan ini sulit untuk terwujud. Untuk itu dalam kesempatan ini penulis ingin menyampaikan ucapan terima kasih yang sebesar-basarnya kepada:

1. Bapak Dr. Eng. Muslim, S.T., M.T selaku Dekan Fakultas Teknik Universitas Islam Riau.
2. Bapak Dr. Arbi Haza Nasution, B.IT(Hons)., M. IT, Selaku Kepala Prodi Teknik Informatika.
2. Ibu Ana Yulianti, S.T., M. Kom Selaku Sekretaris Prodi Teknik Informatika.
3. Bapak Apri Siswanto, S. Kom., M. Kom selaku Pembimbing yang telah banyak membantu penulis memberikan pengarahan dan bimbingan dalam menyelesaikan skripsi ini dengan baik.

4. Bapak Dr. Evizal Abdul Kadir, S.T., M. Eng dan Bapak Yudhi Arta, ST, M.Kom, selaku Dosen Tim Penguji Sidang Tugas Akhir yang telah bersedia memberikan waktu dan sarannya kepada penulis.
5. Seluruh Dosen Teknik Informatika Beserta Staf Tata Usaha.
6. Semua pihak yang telah membantu penyelesaian skripsi ini yang tidak bias penulis sebutkan satu persatu.

Penulis menyadari sepenuhnya bahwa dalam penyusunan laporan skripsi ini masih banyak kekurangan, untuk itu dengan segala kerendahan hati penulis mengharapkan saran dan kritik yang sifatnya membangun guna memperbaiki kekurangan yang terdapat didalam laporan skripsi tersebut.

Akhir kata semoga laporan skripsi ini dapat menambah ilmu pengetahuan dan bermanfaat bagi semua pihak yang membacanya.

Wassalamu'alaikum Wr. Wb.

Pekanbaru, September 2020

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

BAB I
PENDAHULUAN

1.1 Latar Belakang Masalah

Pesan rahasia adalah sebuah pernyataan rahasia yang dibuat oleh seseorang dan ditujukan untuk orang yang dikehendaki. Sangat pentingnya informasi dari sebuah pesan rahasia menyebabkan pesan tersebut tidak sampai kepada penerimanya, melainkan pesan tersebut jatuh ke tangan orang lain yang tidak dikehendaki. Untuk menjaga keamanan informasi pesan, maka salah satu solusinya yaitu dengan ilmu Steganografi.

Steganografi merupakan salah satu teknik yang digunakan dalam pengembangan informasi, yaitu dengan menyembunyikan informasi kedalam media digital dengan metode tertentu agar tidak terlihat perbedaan secara visual antara *file* asli dengan *file* yang telah disisipi informasi, sehingga tidak diketahui oleh pihak lain. Teknik steganografi membutuhkan dua *property*, yaitu media pembawa dan pesan rahasia. Media pembawa yang umumnya digunakan adalah gambar, suara, video, atau teks. Adapun pesan yang disembunyikan dapat berupa artikel, gambar, kode program, atau pesan lain. Media pembawa *audio* dipilih dalam penelitian ini karna *audio* mempunyai kapasitas yang lebih besar dibandingkan dengan berkas teks maupun gambar dan tidak terlalu rumit jika dibandingkan berkas video. Format audio yang digunakan adalah MP3.

Penggunaan MP3 sekarang menjadi sangat tinggi karena pada dasarnya sebagian besar manusia lebih suka melakukan hal-hal yang dapat menghibur. MP3 lebih dipilih untuk digunakan dalam penelitian ini dibandingkan menggunakan file gambar. Dalam melakukan penyembunyian pesan, akan lebih

dipilih format yang lebih sering digunakan sehingga tidak menimbulkan kecurigaan yang terlalu berlebihan. Hal ini dikarenakan lalu lintas pertukaran MP3 di internet menjadi suatu hal yang biasa. Sehingga steganografi menggunakan MP3 merupakan teknik yang baik digunakan untuk mengamankan pesan rahasia melalui media internet.

Pada penelitian ini dibutuhkan metode yang dalam pemrosesan algoritma dengan tingkat keamanan yang tinggi, sehingga mampu mengenkripsi dan deskripsi data tanpa mengubah integritas data tersebut. Berdasarkan latar belakang permasalahan di atas, maka penelitian ini akan menggunakan metode *Parity Coding* dan Enkripsi *Vigenere Cipher*. Kombinasi dari dua algoritma tersebut akan memudahkan dalam pengamanan data. Sehingga penulis menarik sebuah judul yaitu "Analisis Teknik Steganografi Pada Audio MP3 Menggunakan Metode *Parity Coding* dan Enkripsi *Vigenere Cipher*".

1.2 Identifikasi Masalah

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

1. Pengiriman data dalam transmisi di jaringan tidak aman, maka diperlukan teknik steganografi dan enkripsi untuk keamanan data.
2. Menguji proses enkripsi menggunakan metode *Vigenere Cipher* dan Steganografi sebagai bentuk keamanan proses pengiriman data.

1.3 Rumusan Masalah

Berdasarkan uraian latar belakang yang telah dikemukakan, maka rumusan masalah dalam penelitian ini adalah "Bagaimana membuat suatu aplikasi

keamanan data dengan teknik steganografi yang mengkombinasikan Metode *Parity Coding* dan Enkripsi *Vigenere Cipher* menggunakan media audio dengan format MP3?".

1.4 Batasan Masalah

Adapun batasan masalah dari penelitian ini adalah:

1. Wadah yang digunakan untuk menyisipkan data adalah media dalam bentuk audio dengan format MP3.
2. Hasil *file output* disimpan dengan format MP3 Stego.
3. Pada implementasi perangkat lunak, data rahasia di enkripsi terlebih dahulu baru disisipkan pada file audio MP3.

1.5 Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah:

1. Untuk mengimplementasi steganografi dengan metode *Parity Coding* pada dokumen audio dengan format MP3.
2. Untuk menguji seberapa aman data yang di enkripsi pada pesan yang disisipkan menggunakan metode *Vigenere Cipher*.

1.6 Manfaat penelitian

Adapun manfaat dari penelitian ini yaitu diharapkan dapat membantu penggunanya untuk mengirim informasi atau pesan yang bersifat rahasia melalui media pembawa citra audio yang berformatkan MP3 agar sampai ke tangan penerima tanpa menimbulkan kecurigaan pada pihak lain.

BAB II
LANDASAN TEORI

2.1 Studi Kepustakaan

Pada penulisan proposal ini penulis mengambil dan menggunakan acuan kepustakaan sebagai referensi atau pedoman dalam pembuatan proposal skripsi. Adapun penelitian yang berhubungan dengan proposal skripsi ini adalah sebagai berikut:

Studi keperpustakaan pertama adalah berdasarkan penelitian yang dilakukan oleh Indra Jaya Kusuma (2017), yang bertujuan untuk mengamankan pesan dengan cara menyembunyikan *file* ke dalam *file* lainnya sedemikian rupa sehingga orang lain tidak menyadari ada sesuatu didalam *file* tersebut dengan menggunakan metode *parity coding* untuk memecahkan *file* audio menjadi beberapa *region* yang berbeda dan mengenkripsi setiap *bit* dari *file* rahasia yang ingin disisipkan pada sebuah sampel *region* yang berisi *parity bit*.

Studi keperpustakaan 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 dimengerti 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 menyembunyian 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 keperpustakaan ketiga adalah berdasarkan penelitian yang dilakukan oleh Andreas Nicolas Tarigan (2014), Steganografi merupakan ilmu menulis atau menyembunyikan pesan tersembunyi dengan suatu cara sehingga selain si pengirim dan si penerima, tidak ada seorangpun yang mengetahui atau menyadari bahwa pesan tersebut adalah sebuah pesan rahasia. Tujuan dari penelitian ini adalah untuk mengimplementasikan steganografi dengan metode *parity coding* dan enkripsi *caesar cipher* pada *file audio* dengan format MP3 dan dapat dijadikan sebagai aplikasi *alternative* dalam penyembunyian *file* yang aman pada *file audio*.

2.2 Dasar Teori

2.2.1 Kriptografi

Kriptografi berasal dari bahasa Yunani, yaitu *crypto* dan *graphia*. *Crypto* berarti *secret* (rahasia) dan *graphia* berarti *writing* (tulisan). *Cryptography* secara umum adalah ilmu dan seni untuk menjaga kerahasiaan berita. Menurut terminologinya kriptografi adalah ilmu dan seni untuk menjaga keamanan pesan ketika pesan dikirim dari suatu tempat ke tempat yang lain (Ariyus, 2008).

Menurut Indra (2017), ada beberapa komponen yang digunakan dalam sistem kriptografi, yaitu:

a. **Algoritma**, merupakan himpunan aturan matematis yang digunakan dalam enkripsi dan deskripsi.

b. ***Plaintext* (M)**, adalah pesan yang hendak dikirimkan (berisi data asli).

c. ***Ciphertext* (C),** adalah pesan ter-enkrip (tersandi) yang merupakan hasil enkripsi.

d. **Kunci**, adalah suatu bilangan yang dirahasiakan, yang digunakan dalam proses enkripsi dan dekripsi.

e. **Kriptologi**, merupakan studi tentang kriptografi dan kriptanalisis.

f. **Kriptanalisis**, merupakan aksi memecahkan mekanisme kriptografi dengan cara menganalisisnya untuk menemukan kelemahan dari suatu algoritma kriptografi sehingga akhirnya dapat ditemukan kunci atau teks asli.

g. **Enkripsi** (fungsi E), adalah proses pengubahan *plaintext* menjadi *ciphertext*.

h. **Dekripsi** (fungsi D), adalah kebalikan dari enkripsi, yakni mengubah *ciphertext* menjadi *plaintext* sehingga berupa data awal/asli.

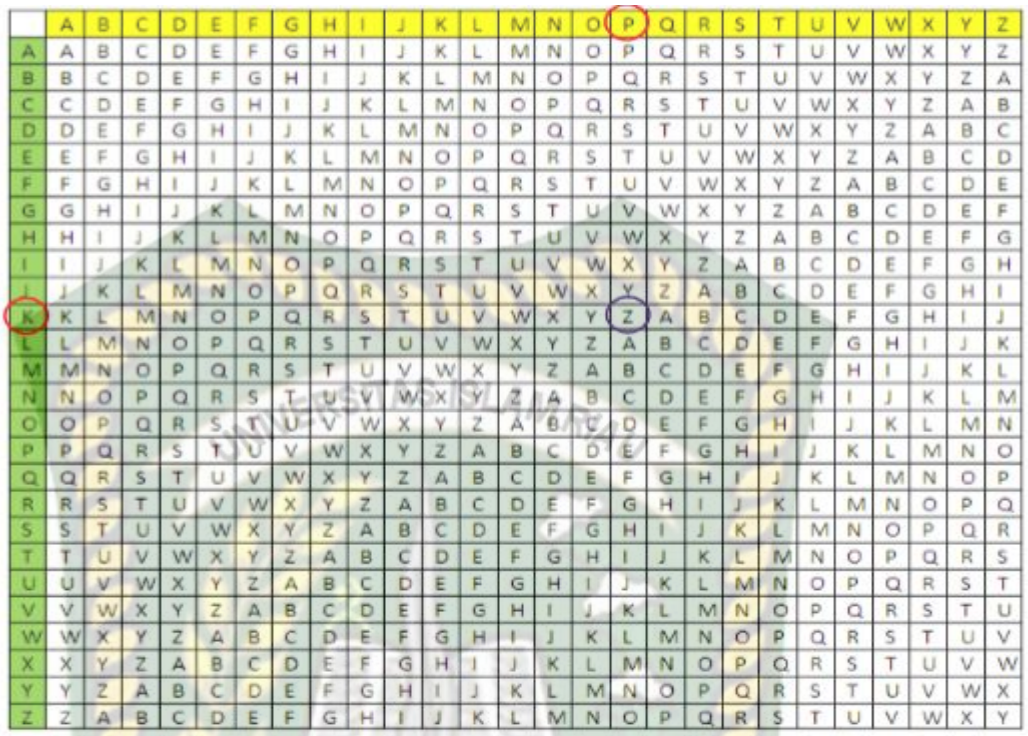
Berikut ini adalah gambaran mekanisme atau cara kerja dalam kriptografi. Perhatikan pada Gambar 2.1

Gambar 2.1 Mekanisme Kriptografi

2.2.2 *Vigenere Cipher*

Vigenere Cipher termasuk kode abjad-majemuk (*polyalphabetic substitution cipher*). Dipublikasikan oleh diplomat (sekaligus seorang kriptologis) Perancis, Blaise de Vigenere pada Abad 16, tahun 1586. Sebenarnya Giovan Batista Belaso telah menggambarkannya untuk pertama kali pada tahun 1553 seperti ditulis di dalam buku *La Cifra del sig*. Algoritma ini baru dikenal luas 200 tahun kemudian dan dinamakan kode *Vigenere*. *Vigenere* juga merupakan pemicu perang sipil di Amerika dan kode *Vigenere* digunakan oleh Tentara Konfiderasi (*Confederete Army*) pada perang Sipil Amerika (*American Civil War*). Kode *Vigenere* berhasil dipecahkan oleh Babbage dan Kasiski pada pertengahan Abad 19 (Ariyus, 2008).

Vigenere Chiper merupakan salah satu algoritma kriptografi klasik untuk menyandikan suatu *plaintext* dengan menggunakan teknik substitusi. *Vigenere Cipher* pada dasarnya cukup rumit untuk dipecahkan. Meskipun begitu, *Vigenere Cipher* tetap memiliki kelemahan. Salah satunya adalah dapat diketahui panjang kuncinya dengan menggunakan metode kasiski. Hal ini disebabkan karena umumnya terdapat frasa yang berulang-ulang pada *ciphertext* yang dihasilkan. Jika diamati lebih lanjut, seiring dengan majunya perkembangan dunia pemrograman, teknik dasar *Vigenere Cipher* dapat sedikit dimodifikasi sehingga tidak hanya mampu melakukan enkripsi terhadap karakter alphabetis saja, namun juga karakter angka dan simbol khusus. Adapun contoh menurut Husodo (2010), misalkan *plaintext* dengan huruf P disandikan dengan kunci K, maka hasil *ciphertext* yang dihasilkan adalah huruf Z. Seperti yang ditunjukan pada gambar 2.2 berikut ini:



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

Gambar 2.2 Bujur Sangkar *Vigenere*

2.2.3 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 (Ariyus, 2006).

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.

2.2.4 Java

Java adalah sebuah bahasa pemrograman *scripting* yang sering digunakan dalam pembuatan aplikasi berbasis *handphone* dan juga dapat digunakan untuk menyediakan akses objek yang disisipkan di aplikasi lain. Java berfungsi sebagai penambah tingkah laku agar *widget* dapat tampil lebih efektif (Garling dan Lestari *dalam* Achmad Fikri Sallaby 2015).

2.2.5 *Parity Coding*

Dalam teknik *parity coding*, sinyal dari berkas audio dipecah menjadi beberapa *region* yang berbeda dan mengenkripsi setiap *bit* dari pesan rahasia yang ingin disisipkan pada sebuah sampel *region* yang berisi *parity bit* (Indra Jaya Kusuma 2017). *Parity bit coding* steganografi pada penelitian ini dilakukan dengan metode penerapan jumlah *parity bit* genap dan ganjil, dimana jika nilai 1 dari biner *bit cover* adalah genap, maka *cover* diasumsikan menyembunyikan *bit* 0, sedangkan jika jumlahnya ganjil, maka diasumsikan menyembunyikan *bit* 1 (I Wayan Candra Winetra, 2017).

2.2.6 *Peak Signal to Noice Ratio* (PSNR)

PSNR adalah sebuah cara untuk mengetahui perubahan yang terjadi antara original file dan stego file dengan cara menghitung perubahan bit yang terjadi antara keduanya. Sebelum perhitungan PSNR dilakukan, disyaratkan untuk

menemukan nilai MSE (*Mean Square Error*). Dengan rumus yang dapat dilihat pada persamaan (2.1) (BhaIshandar dan Gulve *dalam* Muhammad Elan Mustakmal, 2018):

$$\text{MSE} = \frac{1}{MN}\sum_{x=1}^{M}\sum_{y=1}^{N}(Sxy - Cxy)^2 \quad (2.1)$$

Setelah nilai MSE berhasil didapatkan, dilanjutkan dengan pencarian nilai PSNR dengan menghitung nilai maksimal sinyal dibagi dengan nilai MSE. Dengan rumus yang dapat dilihat pada persamaan (2.2) (BhaIshandar dan Gulve *dalam* Muhammad Elan Mustakmal, 2018):

$$PSNR = 10\log_{10}(\frac{255}{\text{MSE}}) \quad (2.2)$$

Dari perhitungan diatas akan di dapatkan nilai PSNR yang kemudian akan digunakan sebagai bahan analisis. Satuan yang digunakan dalam nilai PSNR adalah *decibel* (dB). Nilai PSNR yang baik adalah diatas 30 dB karena jika nilai PSNR berada dibawah 30 dB, maka dapat dipastikan kedua audio tersebut memiliki kesamaan yang sedikit.

2.2.7 *Unified Modelling Language* (UML)

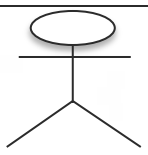
Unified Modeling Language (UML) adalah Bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk menudukung pembangunan sistem (Windu dan Grace *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 *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).

2.2.8 *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 ingin 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, system yang lain atau alat ketika berkomunikasi dengan *use case*
3	*Relationship*	 * *	Penghubung antara objek satu dengan yang lain.

2.2.9 *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

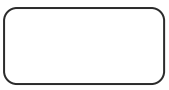
No	SIMBOL	PENJELASAN
1	Class1::**Class**	*Class*, digambarkan sebagai sebuah kotak yang terbagi atas 3 bagian. Bagian atas adalah bagian nama dari *class*. Bagian tengah mendefenisikan property/atribut *class*. Bagian akhir mendefenisikan *method-method* dari sebuah *class*.
2	* *	*Assosiation*, digunakan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

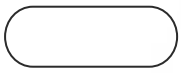
3		*Composition*, jika sebuah *class* tidak bisa berdiri sendiri dan harus merupakan bagian dari *class* yang lain, maka *class* tersebut memiliki relasi *composition* terhadap *class* tempat dia bergantung tersebut. Sebuah *Relationship composition* digambarkan sebagai garis dengan ujung berbentuk jajaran 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.

2.2.10 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

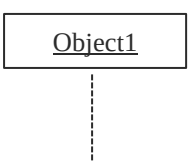
No	SIMBOL	PENJELASAN
1		*Activity*, memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain.

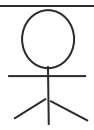
2		*Action, state* dari sistem yang mencerninkan eksekusi dari suatu aksi.
3		*Initial State,* bagaimana objek dibentuk atau diawali.
4		*Final State,* bagaimana objek dibentuk dan diakhiri.
5		*Decision,* digunakan untuk menggambarkan suatu keputusan atau tindakan yang harus diambil pada kondisi tertentu.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

2.2.11 *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.
5	 Message1	*Message*, digambarkan dengan anak panah horizontal antara *Activation Message* mengindikaskan komunikasi antara objek-objek.

BAB III
METODOLOGI PENELITIAN

3.1 Metodologi Penelitian

Metodologi 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. Metodologi juga merupakan analisis teoretis mengenai suatu metode.

3.2 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.

3.2.1 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:

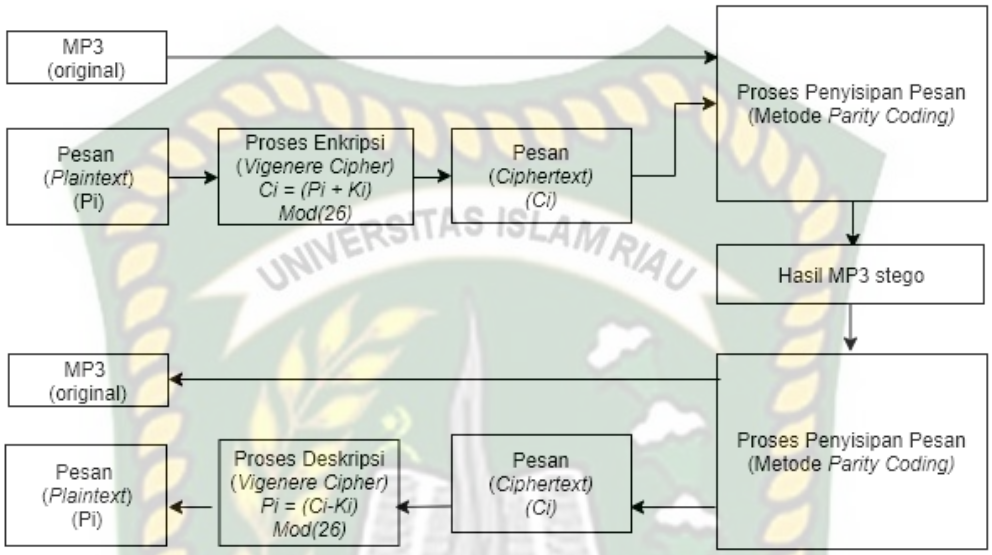
Tabel 3.1 Spesifikasi Hardware dan Software

No	*Hardware* dan *Software*	Spesifikasi	Fungsi
1	Laptop	• *Processor Intel Core* i3-6006U • Ram 4 GB • *Hardisk* 1TB • 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*	• Sistem operasi yang digunakan penulis dalam melakukan penelitian
3	Aplikasi	• *Microsoft Visio 2007* • *Java NetBeans*	• Digunakan untuk membuat diagram, *flowchart* dan diagram lainnya. • Sebagai bahasa pemrograman untuk membangun sebuah aplikasi berbasis desktop

3.3 Usulan Skema Enkripsi

Pada perancangan skema pengenkripsian steganografi audio MP3 terdapat dua konsep yang harus dipertimbangkan sebelum memilih metode mana yang akan dipakai, yaitu format digital audio dan media transmisi dari audio. Oleh karena itu penulis mencoba melakukan pengujian pengenkripsian data tersebut menggunakan metode *parity coding* dan enkripsi *vigenere cipher.* Adapun proses yang harus dilakukan pengguna yaitu, meganti *bit-bit* yang ada didalam audio

MP3 dengan *bit*-bit pesan. Usulan skema yang akan dibangun dapat dilihat pada gambar 3.1.



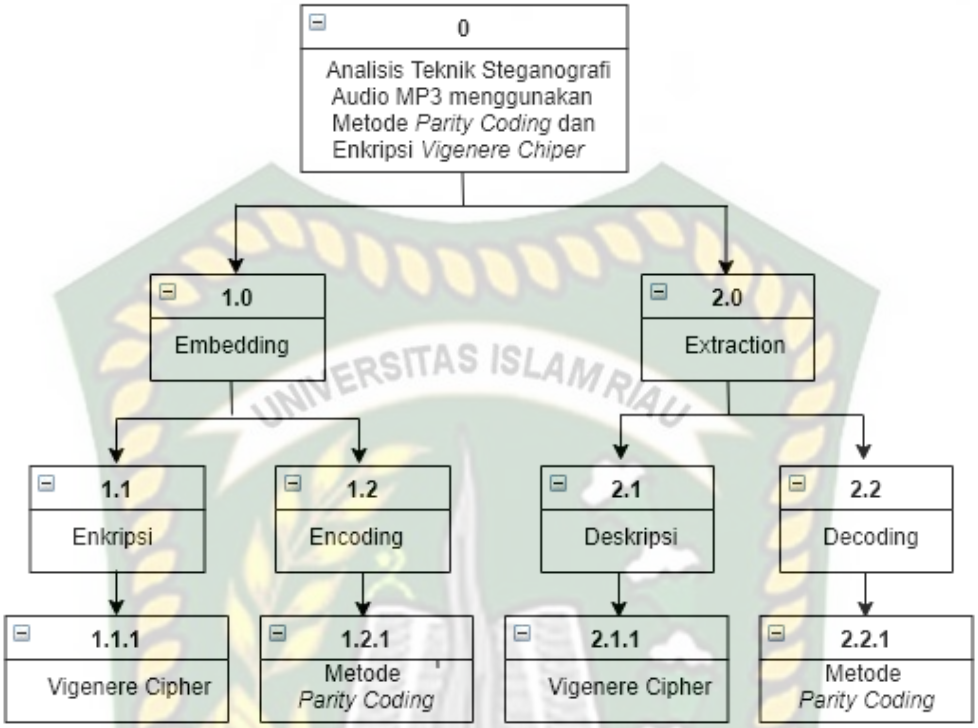
Gambar 3.1 Skema yang akan dibangun

3.4 Pengembangan dan Perancangan Sistem

Pada perancangan pengenkripsian steganografi audio MP3 menggunakan *parity coding* dan enkripsi *viginere cipher* ini terdiri dari 2 proses utama. Pertama proses enkripsi file audio. Kedua ekstraksi atau pengembalian seperti semula atau mendekripsikan file audio yang telah dienkripsikan 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 audio MP3 menggunakan metode *parity coding* dan enkripsi *vigenere cipher*, dapat dilihat pada gambar 3.2.



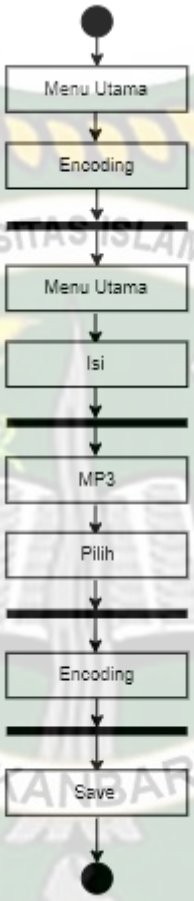
Gambar 3.2 *Hirarchy Chart*

Berdasarkan *Hirarchy Chart* pada gambar 3.2, terdapat 2 proses utama yang terdiri dari proses *Encoding* dan proses *Decoding*. Pada proses *Encoding* akan dilakukan enkripsi file mp3 dengan menggunakan metode *parity coding* dan enkripsi *viginere cipher*. Selanjutnya pada proses *Decoding*, akan dilakukan deskripsi pada file mp3 dengan menggunakan menggunakan metode *parity coding* dan enkripsi *viginere cipher*.

3.4.2 *Activity* Diagram

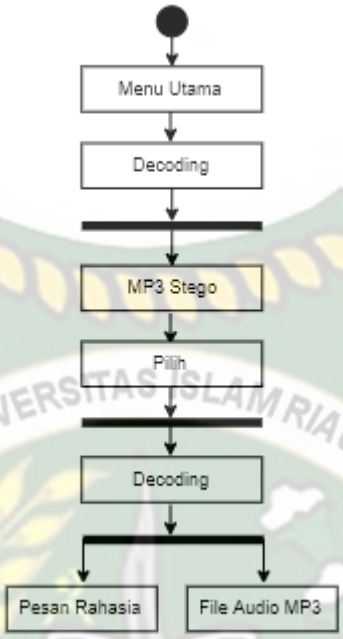
Activity Diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity Diagram* pada sistem ini terdiri

atas 2 bagian, yaitu *Activity Diagram Encoding* dan *Activity Diagram Decoding. Activity Diagram Encoding* dapat dilihat pada gambar 3.3.



Gambar 3.3 *Activity Diagram Encoding*

Pada gambar 3.3 dapat dijelaskan bahwa proses *Encoding* terdiri dari beberapa tahap. Tahap pertama dimulai dari pengirim pesan terlebih dahulu harus menginputkan pesan rahasia dan MP3 yang akan di enkripsi. Selanjutnya yaitu tahap steganografi dengan menggunakan metode *parity coding* dan *vigenere cipher*. MP3 *(MP3 stego)* ini digunakan sebagai media untuk menyisipkan pesan yang akan dijadikan sebagai inputan pada tahap selanjutnya. Adapun untuk tahapan *Activity Diagram Decoding* dapat dilihat pada gambar 3.4

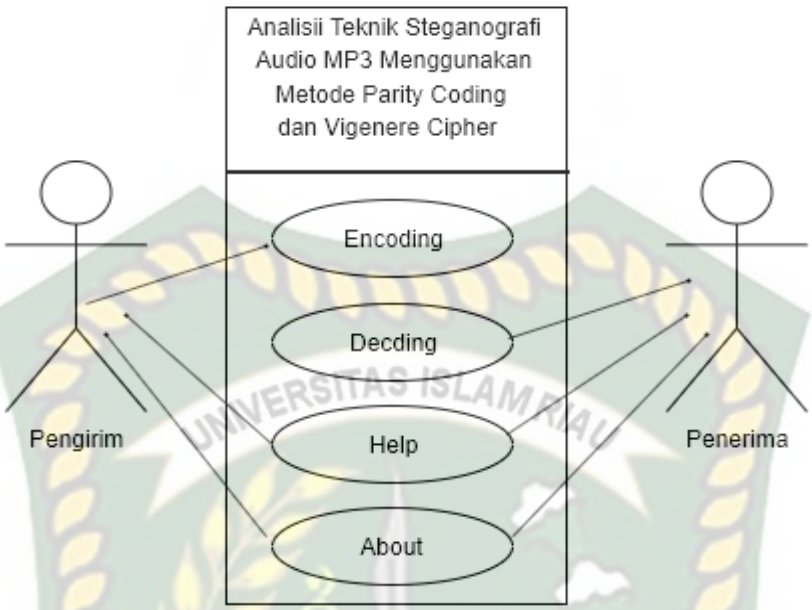


Gambar 3.4 *Activity* Diagram *Decoding*

Pada gambar 3.4 dapat dijelaskan bahwa proses *decoding* terdiri dari beberapa tahap, yaitu tahap pertama yaitu MP3 *stego* (MP3 yang telah disisipkan pesan) yang telah dipilih dan dilakukan proses *decoding* menggunakan metode *parity coding* dan *vigenere cipher*. Dari hasil ekstraksi tersebut akan mengubah *ciphertext* tersebut menjadi *plaintext* agar pesan dapat dibaca oleh penerima pesan.

3.4.3 *Use Case* Diagram

Pada perancangan aplikasi enkripsi teks pada media MP3 menggunakan metode *parity coding* dan *vigenere cipher* terdiri dari proses yang dilakukan oleh pengirim pesan yaitu, proses *encoding* pesan, dan proses yang dilakukan penerimaan pesan yaitu, *extraction* pesan dengan menggunakan metode *parity coding* dan *vigenere cipher*. *User Case* aplikasi yang akan dibangun dapat dilihat pada gambar 3.5 dibawah ini.



Gambar 3.5 *Use Case* Sistem Yang Dibangun

Pada gambar 3.5 diatas dapat dilihat pada aplikasi yang akan dibangun terdiri dari 2 aktor, pertama pengirim pesan dan kedua penerima pesan, dan pada aplikasi ini terdiri dari 4 *case* yaitu, *encoding, decoding, help* dan *about.*

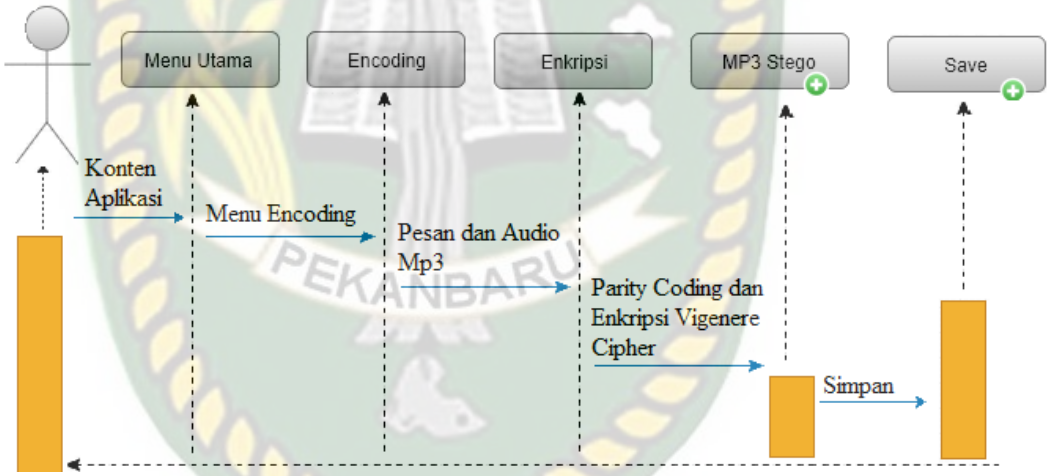
3.4.4 *Class* Diagram

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 3.6 dibawah ini.

Gambar 3.6 *Class Diagram*

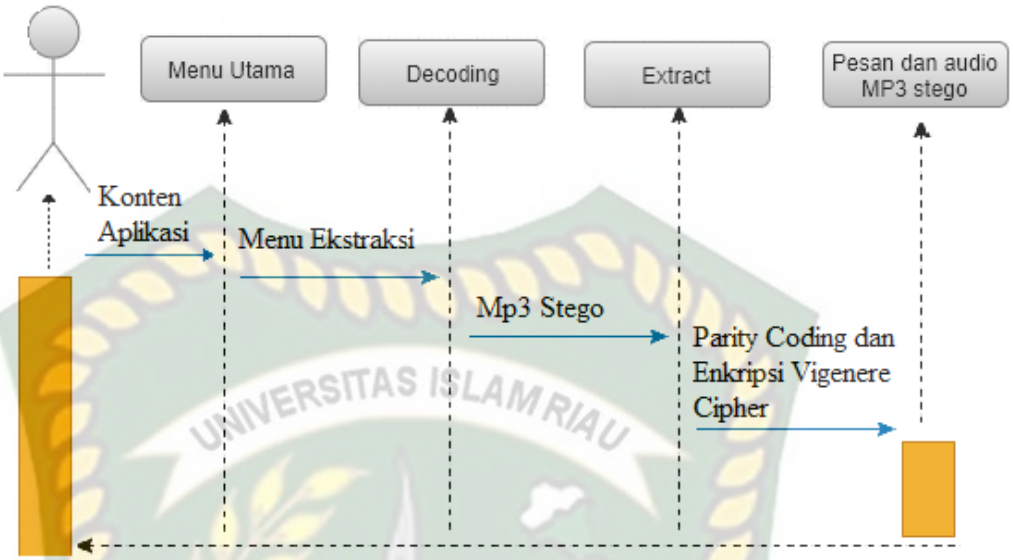
3.4.5 *Sequence* Diagram

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* Pengirim Pesan

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 MP3, kemudian menyimpan MP3 *stego*.

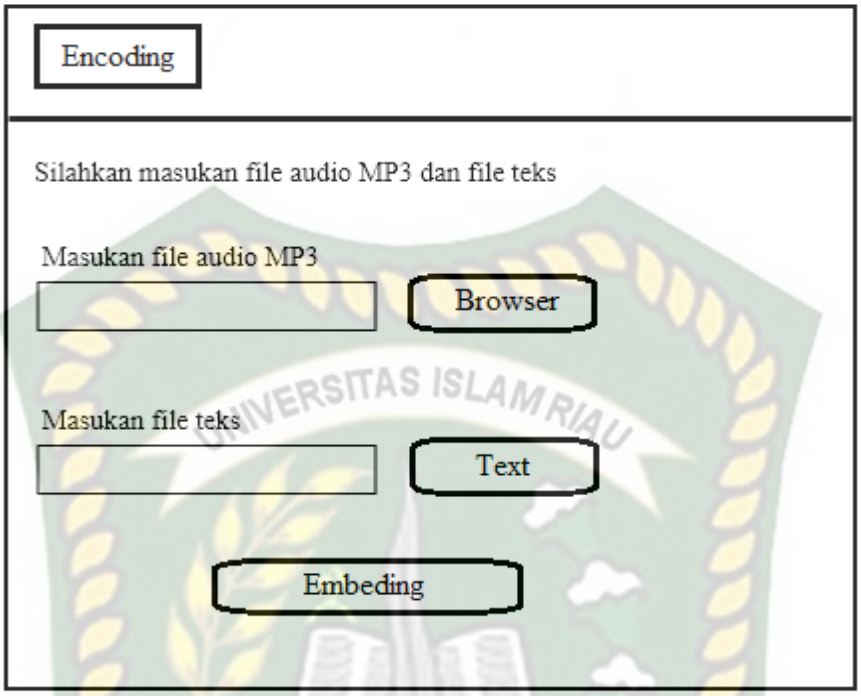


Gambar 3.8 *Sequence Diagram* Penerima Pesan

Berdasarkan gambar 3.8 diatas pembaca pesan dapat dilihat langkah-langkah yang dilakukan pembaca pesan mulai dari menjalankan aplikasi sampai dengan melakukan proses pembacaan pesan yang ada pada MP3 *stego*.

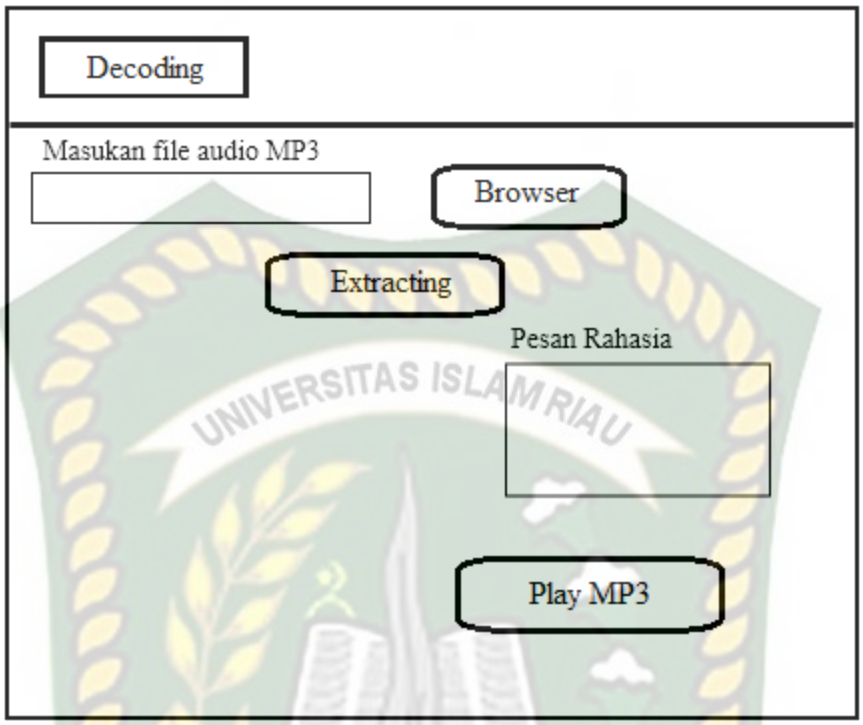
3.5 Perancangan *Input Encoding* dan *Decoding*

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 audio MP3. Tampilan input *encoding* dan *decoding* dapat dilihat pada gambar 3.9 dan 3.10 dibawah ini.



Gambar 3.9 Desain Input *Encoding*

Pada Gambar 3.9 proses desain *input Encoding* dilakukan dengan cara memasukkan *file* audio, memasukkan pesan yang ingin disisipkan kedalam *file* audio tersebut. Setelah di sisipkan pesan rahasia, *file* disimpan dengan cara *Embeding*.

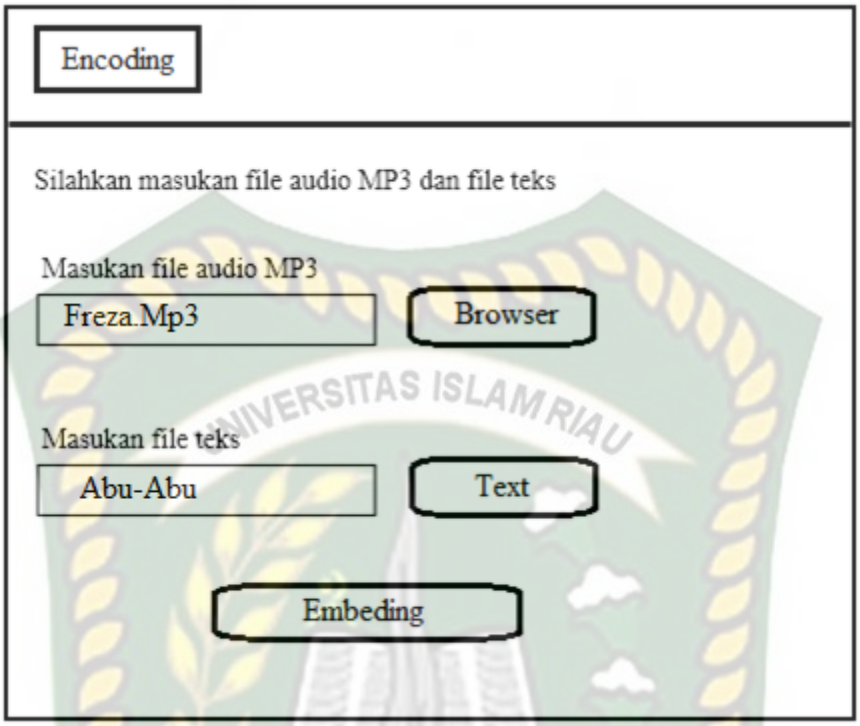


Gambar 3.10 Desain Input *Decoding*

Pada gambar 3.10 Proses desain *input Decoding* dilakukan dengan cara memasukan *file* audio MP3 yang telah di enkripsikan, lalu memasukan kunci agar pesan yang terenkripsi pada *file* audio dapat dilihat pada saat ekstraksi dan *file* audio dapat dijalankan.

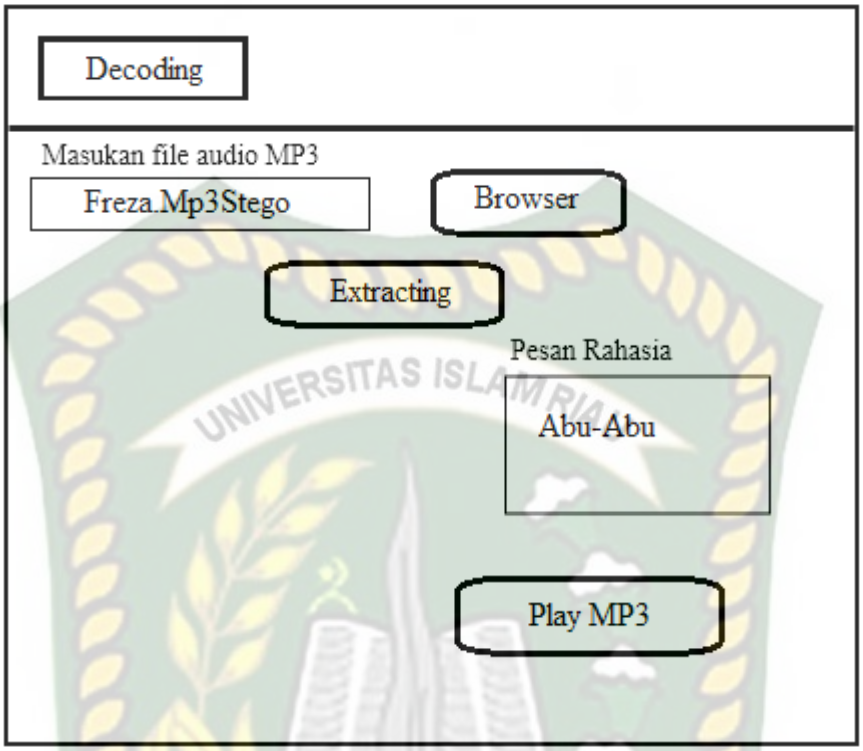
3.6 Perancangan *Output Encoding* dan *Decoding*

Desian *output* merupakan rancangan tampilan *output* atau hasil dari sistem setelah melakukan proses yang terdiri dari *encoding* dan *decoding.* Hasil output merupakan rancangan bentuk tampilan output dari sistem setelah melakukan proses *encoding* dan *decoding* berupa pesan rahasia yang telah disisipkan dalam media MP3. Adapun hasil dari output dapat dilihat pada gambar 3.11 dan 3.12 dibawah ini.



Gambar 3.11 Desain Output Encoding

Pada gambar 3.11 proses desain *output Encoding* dilakukan dengan cara memasukkan *file* audio Mp3, memasukan pesan rahasia, dan disimpan dengan proses *Embeding*.

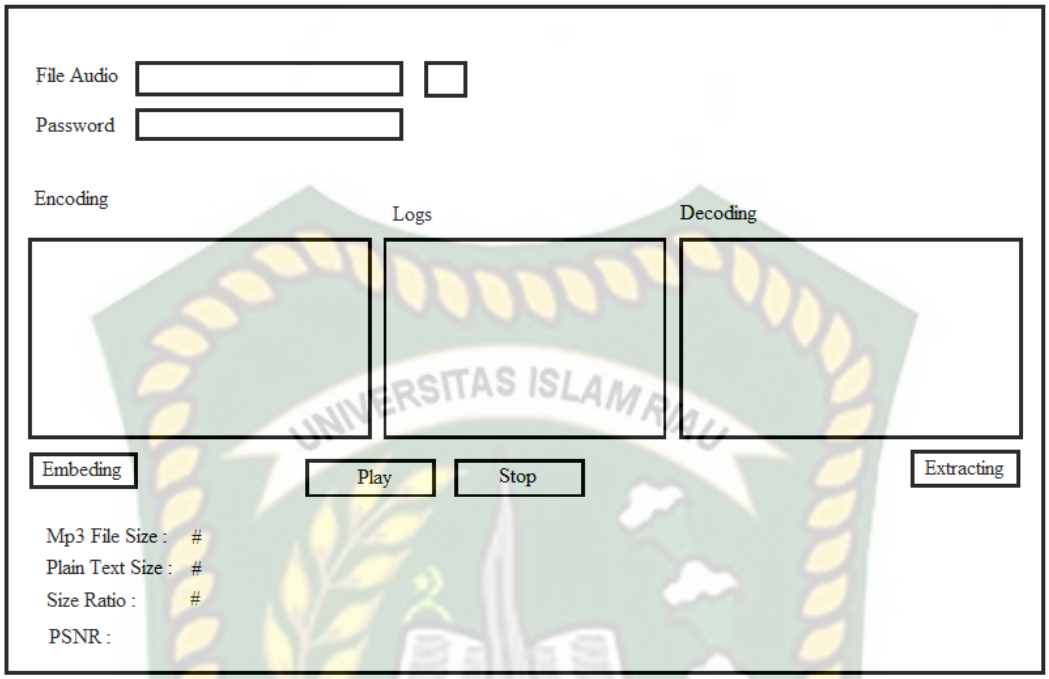


Gambar 3.12 Desain *Output Decoding*

Pada gambar 3.12 proses desain *output decoding* dilakukan dengan cara memasukan *file* audio yang sudah di enkripsi, memasukan kunci agar *file* dapat dilihat dan Mp3 dapat dijalankan setelah dilakukannya proses ekstraksi.

3.7 Desain *Interface* Halaman Utama

Pada halaman utama menampilkan halaman awal pada sistem pertama kali dijalankan. Adapun tampilan halaman utama dapat dilihat pada Gambar 3.13 dibawah ini.



Gambar 3.13 Halaman Utama

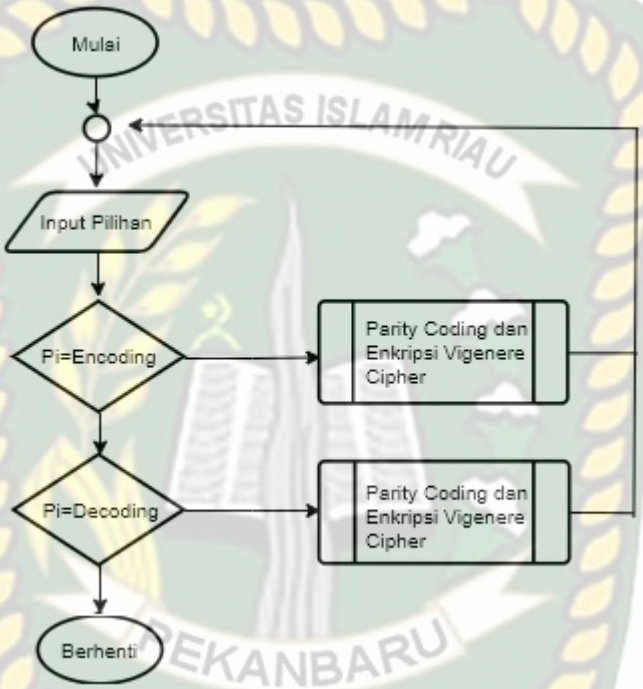
Pada gambar 3.13 diatas terdapat beberapa menu yaitu *encoding, logs, decoding*. *Encoding* merupakan menu untuk menyembunyikan, penyandian pesan dan mengenkripsi pesan ke gambar. *Decoding,* merupakan membaca pesan yang disisipkan. *Logs*, merupakan tempat menampilkan hasil dari proses pengenkripsian pesan

3.8 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.

3.8.1 *Flowchart* Menu Utama

Pada *Flowchart* Menu Utama yang terlihat saat program dijalankan adalah menu utama akan menampilkan menu *encoding* dan *decoding.* Seperti yang dapat dilihat pada gambar 3.14 dibawah ini.



Gambar 3.14 *Flowchart* Menu Utama

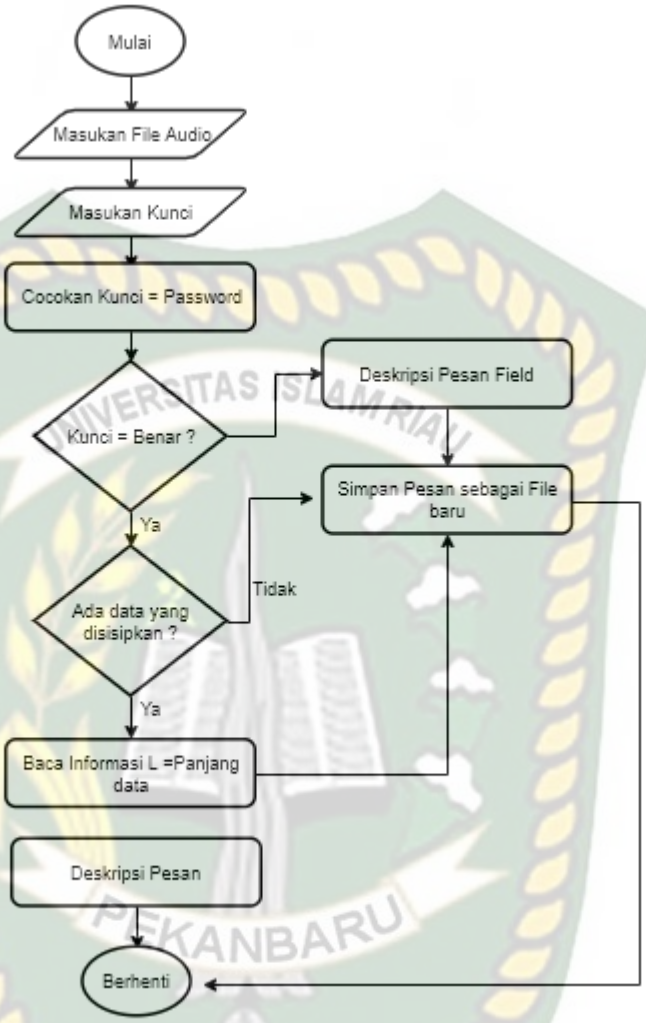
3.8.2 *Flowchart Encoding*

Pada *Flowchart* Menu *Encoding* dapat dilihat, pada saat menu *encoding* maka menampilkan menu *encoding* maka pilih *encoding*. Pada menu *encoding* terdapat beberapa inputan yaitu, pesan, kunci dan MP3. Pada menu *encoding* terdapat proses enkripsi, yang menjadi outputnya *MP3 stego* dan setelah di return pada program akan kembali ke menu utama. Seperti yang dapat dilihat pada Gambar 3.15 dibawah ini.

Gambar 3.15 *Flowchart Encoding*

3.8.3 *Flowchart Decoding*

Pada *Flowchart* Menu *Decoding* dapat dilihat, pada saat menu *Decoding* dipilih menampilkan menu *Decoding.* Pada menu ini terdapat menu *Decoding*, yang menjadi outputnya pesan rahasia dan MP3 yang aslinya, dan setelah direturn program akan kembali ke menu utama. Seperti yang dapat dilihat pada gambar 3.16 dibawah ini.



Gambar 3.16 *Flowchart Decoding*

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 Teknik Steganografi Pada Audio MP3 Menggunakan Metode *Parity Coding* dan Enkripsi *Vigenere Cipher*. Pengujian yang akan dilkaukan 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* pada teknik Steganografi audio MP3 menggunakan metode *Parity Coding* dan Enkripsi *Vigenere Cipher* adalah sebagai berikut dapat dillihat pada tabel 4.1

Tabel 4.1 Pengujian *Black box*

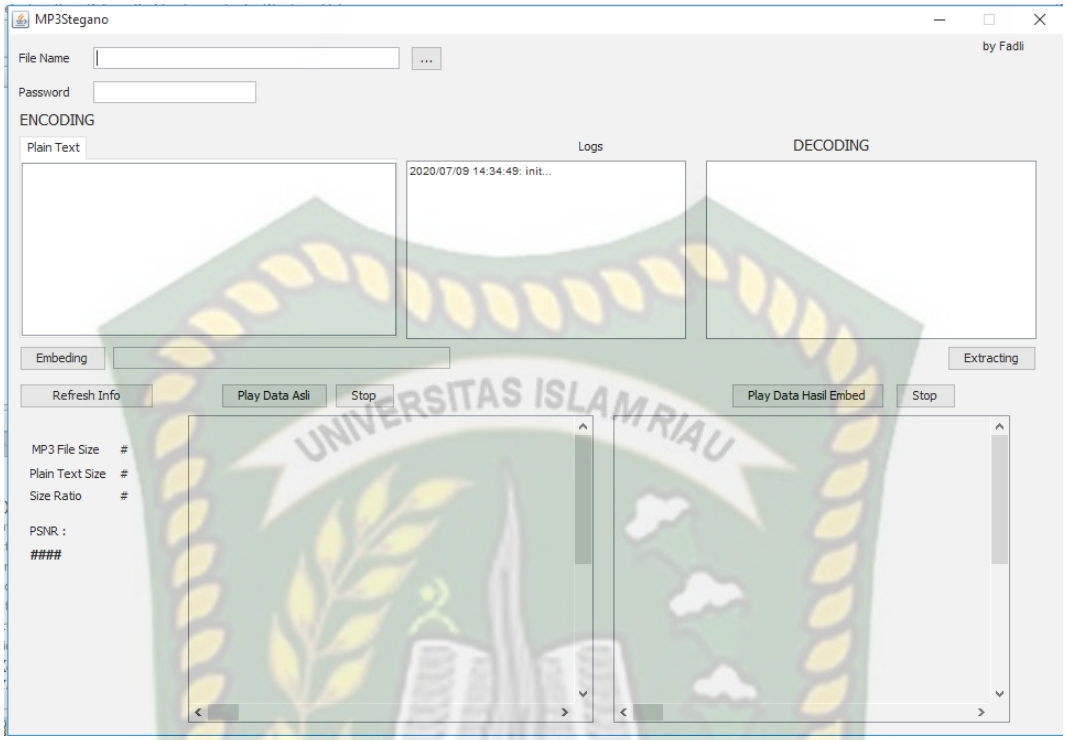
Deskripsi	Prosedur Pengujian	Masukan	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
Encoding	Memasukkan pesan baru yang akan diuji	Pesan (*plaintext)*, kunci, dan MP3 original	Memunculkan hasil *enkripsi*	Sesuai harapan	Berhasil
Decoding	Memasukkan stego audio yang telah di lakukan sebelumnya	Stego audio, kunci	Memunculkan hasil *extraction*	Sesuai harapan	Berhasil
Logs		-	Memuncul		

	-		kan proses dan hasil pengenkripsian pesan		Berhasil
Play Audio	Memutar lagu asli, lagu yang di enkripsi, dan lagu yang di deskripsi	-	Memutar lagu asli, lagu yang di enkripsi, dan lagu yang dideskripsi	Sesuai harapan	Berhasil
Stop	Proses pemutaran lagu dihentikan	-	Menghentikan proses dari pemutaran lagu asli, lagu yang di enkripsi dan lagu yang dideskripsi	Sesuai harapan	Berhasil

4.2 Penjelasan Sistem

4.2.1 *Form* Menu Utama

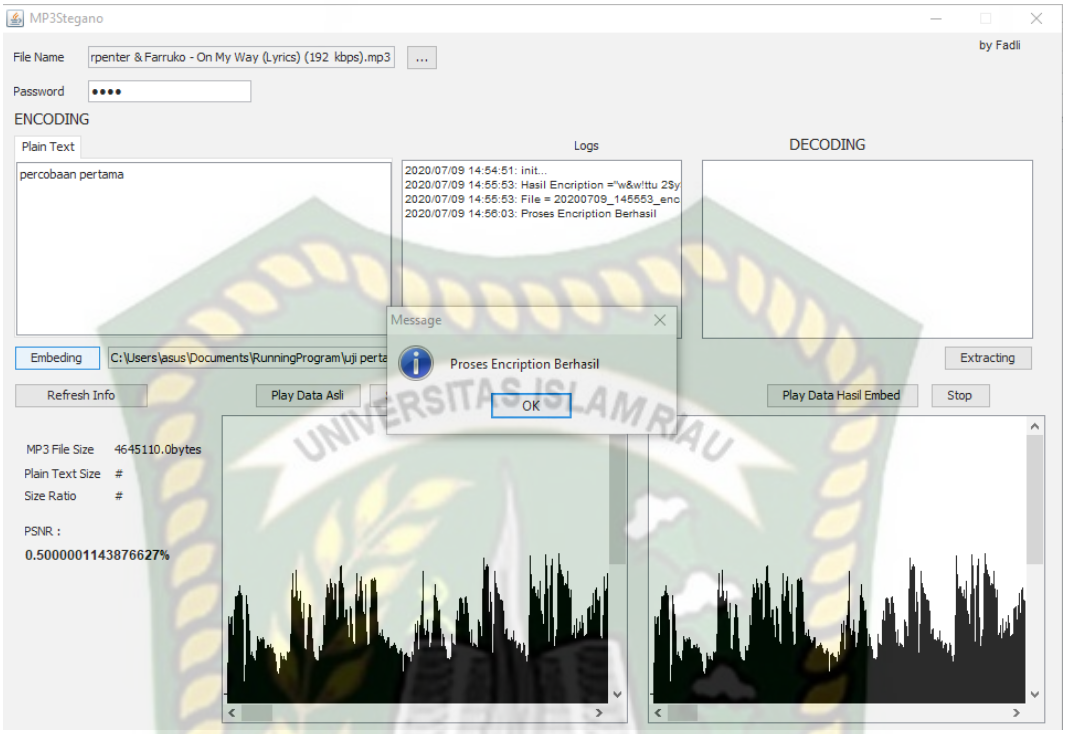
Pada halaman menu utama terdapat menu-menu pilihan yang memiliki fungsi masing-masing, yaitu menu *Encoding*, *Decoding*, *Logs*. Menu *encoding* merupakan tempat melakukan pengenkripsian pesan dengan Mp3. Menu *decoding* merupakan tempat pemisah atau pengekstrasian pesan yang telah di enkrpsi dengan Mp3. Dan pada menu *Logs* merupakan tempat menampilkan proses dan hasil dari pengenkripsian pesan dengan Mp3. Untuk lebih jelasnya dapat dilihat pada gambar 4.1



Gambar 4.1 *Form* Menu Utama

4.2.2 *Form Encoding*

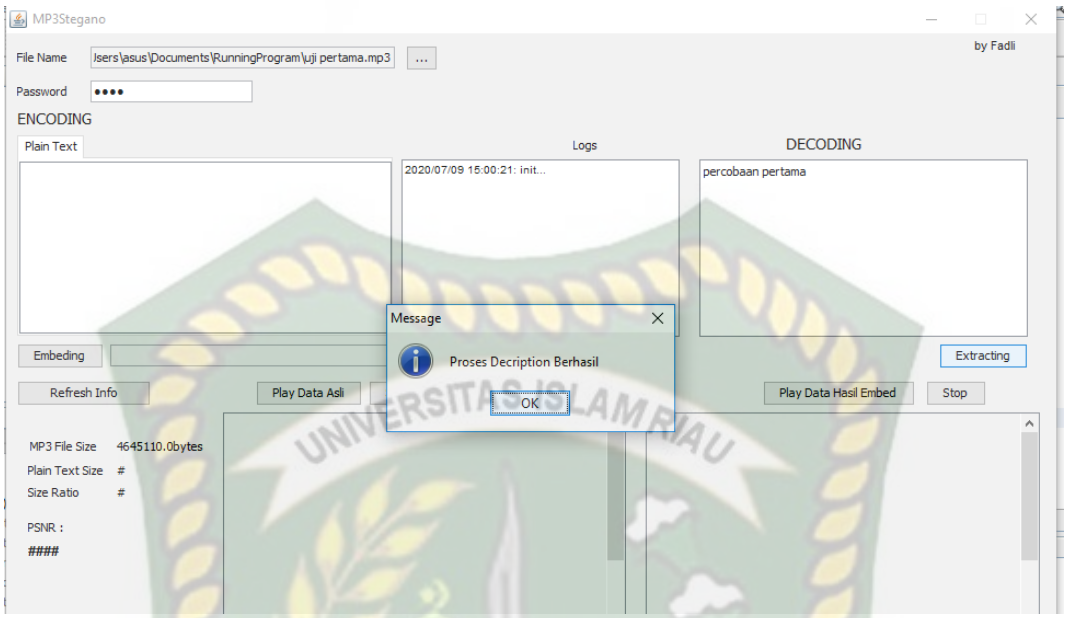
Form Encoding merupakan proses enkripsi pesan. Hasil enkripsi pesan tersebut berupa *chipertext* dan disembunyikan ke dalam Mp3 yang sudah dipilih. Untuk melakukan enkripsi pesan, maka langkah yang harus dilakukan adalah memilih lagu yang ingin dijadikan wadah sebagai tempat menyisipkan pesan, memasukan sandi penyisipan pesan, serta memasukan pesan yang ingin disisipkan, lalu menekan tombol *embedding*. Proses enkripsi pesan dapat dilihat pada gambar 4.2.



Gambar 4.2 *Form* Menu *Encoding*

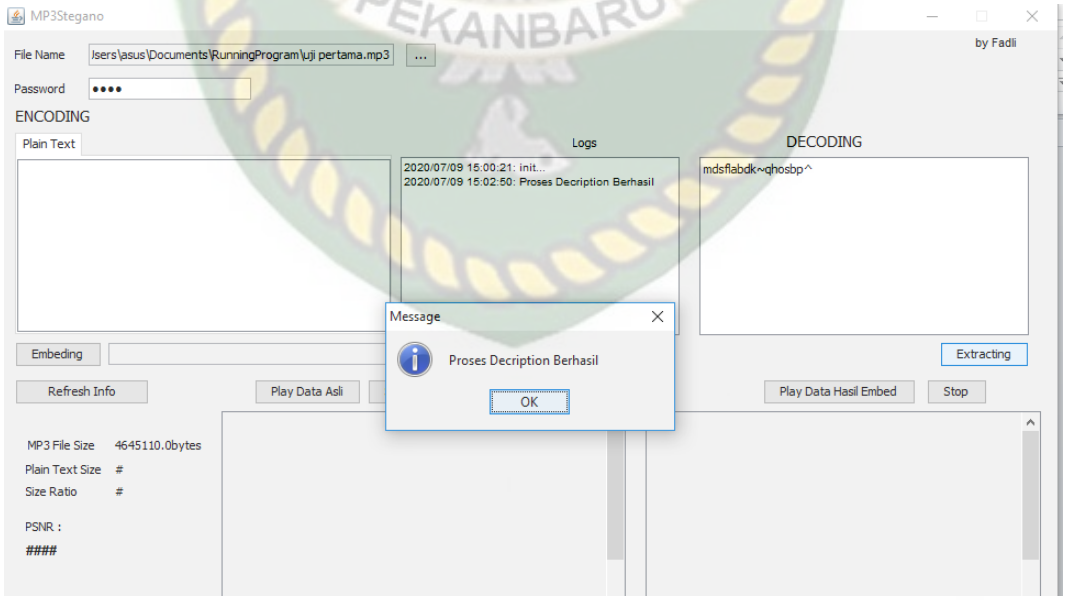
4.2.3 *Form Decoding*

Form decoding merupakan proses mengekstrak pesan yang telah di enkripsi sebelumnya. Sehingga pesan yang telah dienkripsi dapat dilihat dan Mp3 yang digunakan sebagai wadah penyisipan pesan bisa diputar. Maka langkah yang harus dilakukan adalah memilih lagu yang sudah di enkripsikan, memasukan sandi pada saat *embedding*, dan menekan tombol *extraction*. Proses *extraction* dapat dilihat pada gambar 4.3.



Gambar 4. 3 *Form Extraction*

Dari gambar 4.3 diatas, apabila salah memasukan password pada proses mengekstrak pesan, maka isi pesan yang sudah dienkripsi tersebut berubah. Proses tersebut dapat dilihat pada gambar 4.4.



Gambar 4.4 Ekstraksi pesan dengan memasukan password yang salah

4.3 *Source Code* untuk Membangun Implementasi

Dalam proses implementasi aplikasi audio steganografi ini, dilakukan proses konversi dari teori dan rancangan kedalam aplikasi java. Berikut ini *source code* yang digunakan dalam proses konversi sehingga teori dan rancangan yang telah dibuat dapat berjalan pada aplikasi java.

4.3.1 Proses *Encoding*

```
private void substitutionEncode(String MP3_InputFile, String strMsg, String password) throws IOException, Exception {
    String MP3_Ext = "mp3";
    Vigenere vig = new Vigenere();
    String MP3_OutputFile;
    String encryptedMsg = vig.Enc(strMsg, password);

    JFileChooser fileChooser = new JFileChooser();
    FileNameExtensionFilter pFilter = new FileNameExtensionFilter("MP3 FileName", MP3_Ext);
    fileChooser.setFileFilter(pFilter);
    int status = fileChooser.showSaveDialog(this);

    if (status == JFileChooser.APPROVE_OPTION) {
        File selectedFile = fileChooser.getSelectedFile();

        try {
            String fileName = selectedFile.getCanonicalPath();
            if (!fileName.endsWith(MP3_Ext)) {
                selectedFile = new File(fileName + "." + MP3_Ext);
            }

            MP3 mp3 = new MP3(MP3_InputFile);
            Message msg = new Message(encryptedMsg);
            mp3.stega(msg, parity);

            mp3.toMP3(selectedFile.getCanonicalPath());
                MP3_OutputFile = selectedFile.toString();
                txtFileNameSim.setText(selectedFile.toString());


               logs("Hasil Encription =" + encryptedMsg);
               //txtPlainText.setText(txtPlainText.getText());

                //write to file
                DateFormat dateFormat = new SimpleDateFormat("yyyyMMdd_HHmmss");
                Date date = new Date();

                FileWriter writer = new FileWriter(dateFormat.format(date)+"_enc.txt", true);
                writer.write(encryptedMsg);
                writer.close();
                logs("File = "+dateFormat.format(date)+"_enc.txt");

                //conver to wat to analyze sound
                Converter c=new Converter();
                c.convert(MP3_InputFile,MP3_InputFile+".wav");
                c.convert(MP3_OutputFile,MP3_OutputFile+".wav");

                Wave w1 = new Wave(MP3_InputFile+".wav");
                Wave w2 = new Wave(MP3_OutputFile+".wav");
```

Gambar 4.5 *Source code* proses *encoding*

Gambar 4.5 merupakan *source code* proses pengenkripsian pesan dimulai dari memasukan lagu, *plaintext*, dan juga *password*. Sebelum audio dijadikan wadah penyisipan pesan, audio berformatkan Mp3 tersebut di *converter* terlebih dahulu ke audio yang berformatkan wav.

```
GraphicRender gr = new GraphicRender();

gr.renderWaveform(w1, "w1_wf.png");
gr.renderWaveform(w2, "w2_wf.png");

File src1 = new File("w1_wf.png");
File dst1 = new File("src\\ui\\w1_wf.png");
copyFileUsingStream(src1,dst1);

File src2 = new File("w2_wf.png");
File dst2 = new File("src\\ui\\w2_wf.png");
copyFileUsingStream(src2,dst2);

jLabel10.setVisible(true);
jLabel11.setVisible(true);

double w1size = get_size(txtFileName.getText()+".wav");
double w2size = get_size(txtFileNameSim.getText()+".wav") + encryptedMsg.length();
double w1w2size = w2size/(w1size+w2size);
lblMaxAllowedSize.setText(String.valueOf(w1w2size) + "%");
//compare images

BufferedImage img1 = ImageIO.read(new File("w1_wf.png"));
BufferedImage img2 = ImageIO.read(new File("w2_wf.png"));
```

Gambar 4.6 *Source code* menampilkan grafik periodogram

Gambar 4.6 merupakan proses untuk menampilkan grafik periodogram. Bertujuan untuk melihat perbedaan hasil dari file yang telah di enkripsi dan belum di enkripsi.

```
public String Enc(String plainText, String key) {

    String cipherText = "";
    int keyIndex = 0;

    for (int ptextIndex = 0; ptextIndex < plainText.length(); ptextIndex++) {
        char pChar = plainText.charAt(ptextIndex);
        int asciiVal = (int) pChar;


        if (asciiVal < 32 || asciiVal > 126) {
            cipherText += pChar;
            continue;
        }
        int basicPlainTextValue = ((int) pChar - 32);

        char kChar = key.charAt(keyIndex);
        int basicKeyValue = ((int) kChar) - 32;
        int tableEntry = vigenereTable[basicPlainTextValue][basicKeyValue];
        char cChar = (char) (tableEntry + 32);
        cipherText += cChar;
        keyIndex++;

        if (keyIndex == key.length()) {
            keyIndex = 0;
        }
    }
    return cipherText;
}
```

Gambar 4.7 *Source code encoding vigenere cipher*

Gambar merupakan *source code* proses menyandikan suatu *plaintext*. Adapun proses penyandiannya adalah memodifikasikan karakter pada *plaintext* dan *password* dengan nilai *ASCII* menjadi *ciphertext*.

4.3.2 Proses *Decoding*

```
private void substitutionDecode(String MP3_FileName, String password) throws IOException, Exception {

    //String TXT_Ext = "txt";
    Vigenere vig = new Vigenere();

    MP3 mp3 = new MP3(MP3_FileName);
    String strMsg = mp3.decoder(parity);
    txtPlainTextDec.setText(vig.Dec(strMsg, password));
    JOptionPane.showMessageDialog(null, "Proses Decription Berhasil");
    logs("Proses Decription Berhasil");


}
```

Gambar 4.8 *Source code* proses *Decoding*

Gambar merupakan *source code* proses pendeskripsian pesan. Bertujuan untuk merubah *ciphertext* menjadi *plaintext*.

```
public String Dec(String cipherText, String key) {

    String plainText = "";
    int keyIndex = 0;

    for (int ctextIndex = 0; ctextIndex < cipherText.length(); ctextIndex++) {
        char cChar = cipherText.charAt(ctextIndex);
        int asciiVal = (int) cChar;

        if (asciiVal < 32 || asciiVal > 126) {
            plainText += cChar;
            continue;
        }
        int basiccipherTextValue = ((int) cChar) - 32;
        char kChar = key.charAt(keyIndex);
        int basicKeyValue = ((int) kChar) - 32;
        for (int pIndex = 0; pIndex < tableColumnSize; pIndex++) {
            if (vigenereTable[basicKeyValue][pIndex] == basiccipherTextValue) {
                char potpChar = (char) (pIndex + 32);
                plainText += potpChar;}
        }
        keyIndex++;

        if (keyIndex == key.length()) {
            keyIndex = 0;
        }
    }
    return plainText;
}
```

Gambar 4.9 *Source code decoding vigenere cipher*

Gambar 4.9 merupakan *source code* yang digunakan untuk merubah *chipertext* menjadi *plaintext*, setelah karakter pada *ciphertext* tersebut dimodifikasi kembali dengan nilai *ASCII*.

4.4 Uji Perbandingan dan Kesimpulan Hasil Pengujian *Black Box*

Untuk melakukan uji perbandingan dalam Teknik Steganografi pada Audio Mp3 menggunakan Metode *Parity Coding* dan Enkripsi *Vigenere Cipher*, penulis memasukan beberapa karakter sebagai media inputan *plaintext*. Untuk *plaintext*, karakter yang dimasukan adalah “percobaan pengujian”. Untuk media inputan

password, karakter yang digunakan adalah "test". Adapun tabel perbandingan dari *file* audio yang belum dan sudah di enkripsi adalah sebagai berikut:

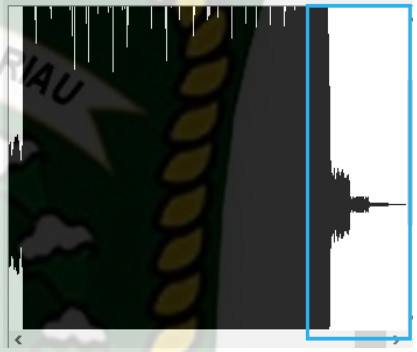
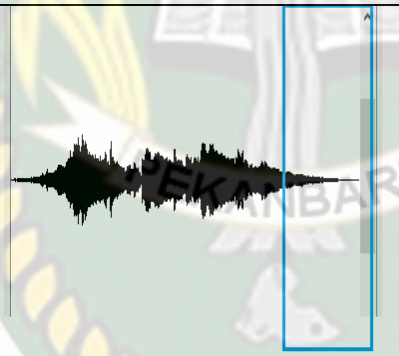
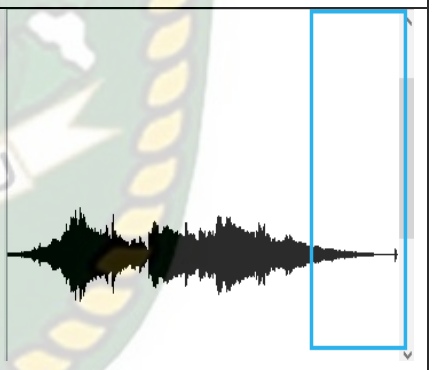
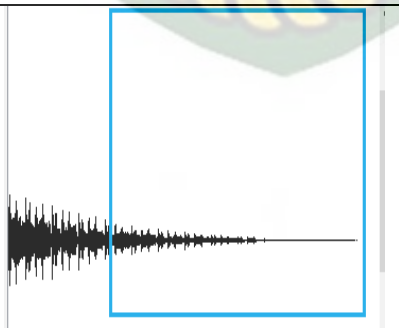
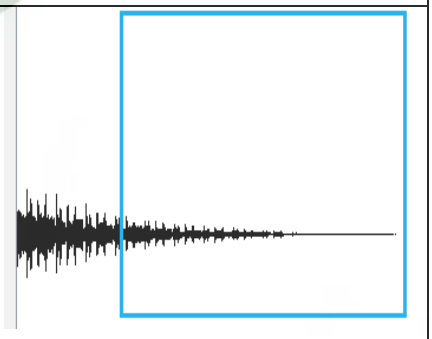
Tabel 4.2 Selisih Ukuran *file* Audio

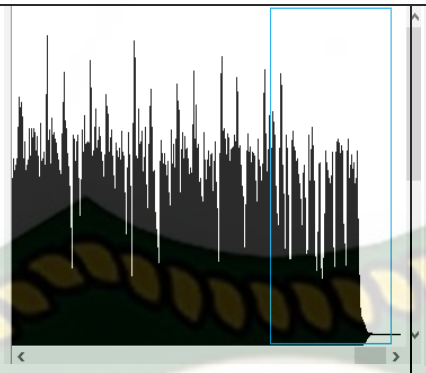
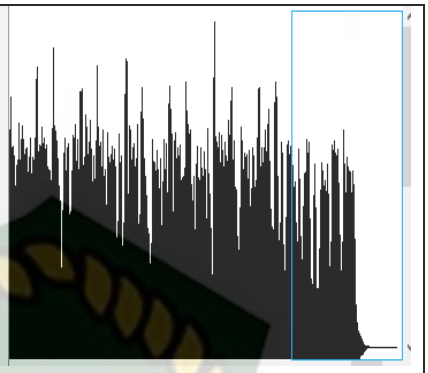
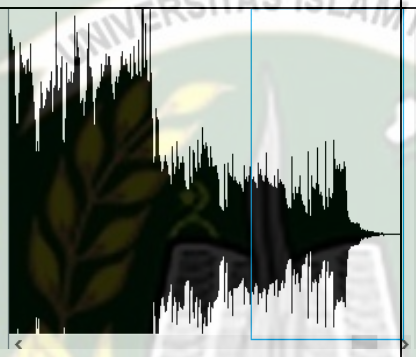
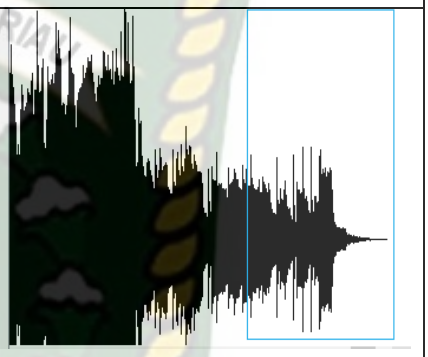
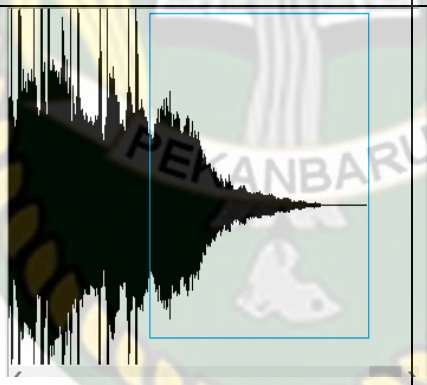
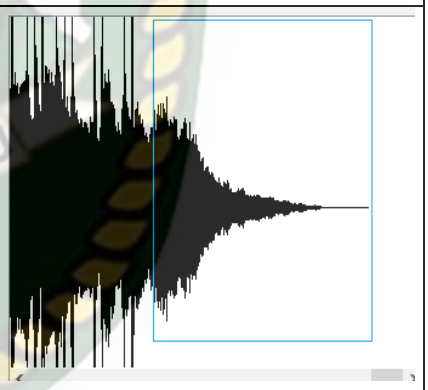
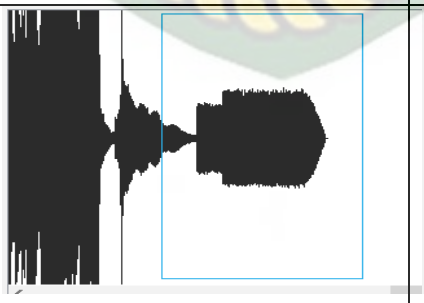
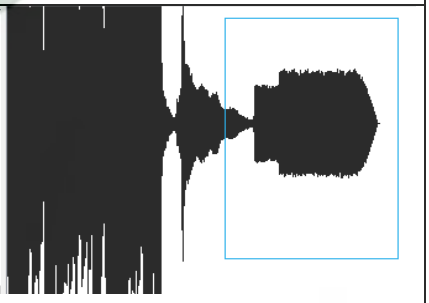
No	*File* Audio	Ukuran Audio (*byte*)	Stego Audio	Ukuran Stego Audio (*byte*)
1	Yellow Claw - Till it Hurts.wav	33.297.452	Uji 1.wav	33.292.844
2	Sia - The Greatest.wav	37.218.860	Uji 2.wav	37.209.644
3	Shawn Mendes - If I *cant* Have you.wav	33.689.132	Uji 3.wav	33.679.916
4	Freza - Abu - abu.wav	43.937.324	Uji 4.wav	43.928.108
5	Pamungkas - Monolog.wav	35.477.036	Uji 5.wav	35.467.820
6	Naif - Piknik 72	38.790.188	Uji 6.wav	38.780.972
7	Hindia - Rumah ke rumah.wav	48.964.652	Uji 7.wav	48.950.828
8	Alan Walker- Alone Pt2	31.592.492	Uji 8.wav	31.578.668
9	One Republic - Counting Star.wav	49.766.444	Uji 9.wav	49.757.228
10	Maroon 5 - Memories.wav	33.661.484	Uji 10.wav	33.652.268

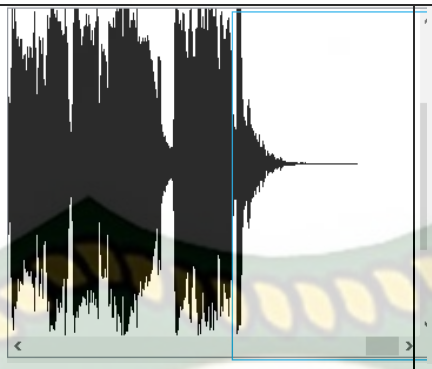
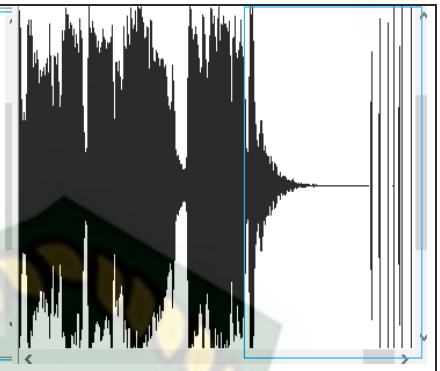
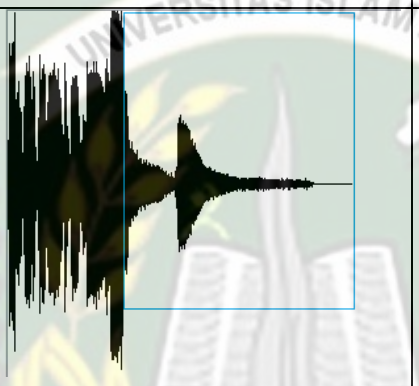
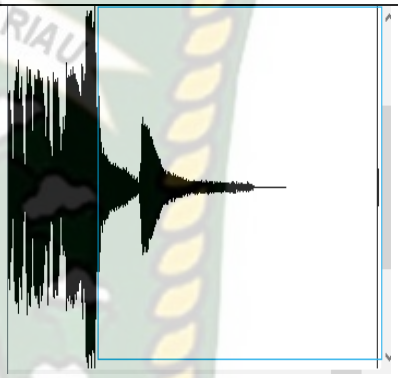
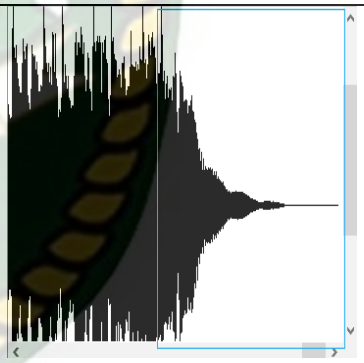
Berdasarkan tabel diatas, terdapat perubahan data pada *file* audio berformat WAV setelah proses *embedding* dilakukan. Dapat dilihat bahwa selisih ukuran

dari *file* asli dengan *file* yang sudah di enkripsikan pesan memiliki pengurangan 4,6 kb dari *file* aslinya. Adapun perubahan data yang terjadi setelah di enkripsikan pesan, dapat dilihat pada table dibawah ini:

Tabel 4.3 Perbandingan hasil periodogram

No	Nama file audio	Periodogram sebelum di enkripsi pesan	Periodogram setelah di enkripsi pesan
1	Yellow Claw - Till it Hurts.wav		
2	Sia - The Greatest. wav		
3	Shawn Mendes - If I cant Have you.wav		

4	Freza - Abu - abu.wav		
5	Pamungkas - Monolog.wav		
6	Naif - Piknik 72		
7	Hindia - Rumah ke rumah.wav		

8	Alan Walker-Alone Pt2		
9	One Republic - Counting Star.wav		
10	Maroon 5 - Memories.wav		

Berdasarkan tabel diatas, perubahan data terjadi pada bagian ujung akhir file audio. Dari hasil pengujian Teknik Steganografi Audio MP3 menggunakan Metode *Parity Coding* dan Enkripsi *Vigenere Cipher* untuk keamanan data teks yang dilakukan dengan metode *black box* maka dapat diperoleh kesimpulan sebagai berikut:

1. Dari seluruh pengujian yang dilakukan pada setiap bagian sistem ini semuanya dapat berfungsi dan berjalan dengan baik.
2. Pengujian fitur yang terdapat pada sistem dapat berjalan dengan normal dan sesuai dengan output yang diharapkan.
3. Pada proses pengenkripsian pesan, terjadi proses *convert* dari *file* Mp3 ke *file* WAV.

4.5 Kesimpulan Hasil Implementasi

Pengujian sistem dari Teknik Steganografi pada Audio Mp3 ini juga dilakukan dari sisi pengguna. Pada pengujian ini dibuat dengan 5 pertanyaan yang disebarkan kepada 20 responden (Mahasiswa dan Masyarakat). Adapun lima pertanyaan tersebut adalah sebagai berikut:

1. Bagaimana pendapat anda mengenai desain tampilan Aplikasi ini?
2. Apakah Aplikasi mudah digunakan (*User Friendly*)?
3. Apakah bahasa yang digunakan dalam aplikasi ini dapat dimengerti dengan baik?
4. Apakah fitur-fitur yang dibangun didalam sistem mudah digunakan?
5. Apakah Teknik Steganografi pada Audio MP3 ini dapat menjaga keamanan data dengan baik?

Dari pertanyaan-pertanyaan diatas, maka hasil jawaban atau tanggapan dari responden terhadap kinerja dari sistem berdasarkan pertanyaan yang diajukan adalah sebagai berikut:

Tabel 4.4 Hasil Jawaban dari Responden

No	Keterangan	Responden		
		Setuju	Cukup	Tidak Setuju
1	Bagaimana pendapat anda mengenai desain tampilan aplikasi ini?	12	4	4
2	Apakah Aplikasi mudah digunakan (User Friendly)?	12	8	0
3	Apakah bahasa yang digunakan dalam aplikasi ini dapat dimengerti dengan baik?	16	3	1
4	Apakah fitur-fitur yang dibangun pada sitem mudah digunakan?	15	5	0
5	Apakah Teknik Steganografi pada Audio MP3 ini dapat menjaga keamanan data dengan baik?	18	2	0

Berdasarkan hasil kuisioner mahasiswa/i dan masyarakat tersebut maka dapat disimpulkan dengan menggunakan skala likert yang telah dimodifikasi, yaitu responden memilih 3 jawaban yang tersedia dengan keterangan, yakni Sangat Baik, Baik dan Cukup. Berdasarkan Pengujian Teknik Steganografi pada Mp3 menggunakan Metode Parity Coding dan Enkripsi Vigenere Cipher ini memiliki persentase sebagai berikut:

Tabel 4.5 Hasil Nilai Presentase Tiap Pertanyaan Kuisioner

No	Keterangan	Jumlah Presentase Responden		
		Setuju	Cukup	Tidak Setuju
1	Bagaimana pendapat anda mengenai desain tampilan aplikasi ini?	60%	20%	20%
2	Apakah Aplikasi mudah digunakan (User Friendly)?	60%	40%	0%
3	Apakah bahasa yang digunakan dalam aplikasi ini dapat dimengerti dengan baik?	80%	15%	5%
4	Apakah fitur-fitur yang dibangun	75%	25%	0%

	pada sitem mudah digunakan?			
5	Apakah Teknik Steganografi pada Audio MP3 ini dapat menjaga keamanan data dengan baik?	90%	10%	0%
Total		73%	22%	5%

Dari hasil persentase tabel diatas, yang didasarkan pada 5 pertanyaan yang diajukan secara langsung oleh penulis kepada 20 responden (Mahasiswa dan Masyarakat), dapat diambil kesimpulan bahwa Teknik Steganografi pada Audio Mp3 menggunakan Metode Parity Coding dan Enkripsi Vigenere Cipher ini memiliki performance setuju dengan nilai (60% + 60% + 80% + 75% + 90%)/5 = 73%, jadi persentase rata-rata terbesar 73%, sehingga sistem ini dapat diimplementasikan.

BAB V
KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil analisa dan pengujian pada bab sebelumnya dalam Teknik Steganografi pada Audio Mp3 menggunakan Metode Parity Coding dan Enkripsi Vigenere Cipher, maka dapat ditarik beberapa kesimpulan yaitu:

1. Dengan adanya Teknik Steganografi dapat membantu menjaga kerahasiaan sebuah pesan sehingga sampai ke tangan orang yang berhak.
2. Pada Teknik Steganografi yang diterapkan ini mampu digunakan untuk menyembunyikan penanda hak cipta pada karya digital.
3. Pada Aplikasi yang dibangun ini output yang dihasilkan terdapat dua bentuk format audio yang berbeda yaitu format Mp3 dan WAV.
4. Dari kedua format tersebut memiliki perbedaan ukuran pada saat penyimpanan file yang sudah di enkripsi. Selisih ukuran file audio Mp3 dengan file audio WAV bisa mencapai 35 Mb.
5. Untuk hasil perbandingan nilai PSNR file audio, diambil dari format audio WAV.

5.2 Saran

Adapun saran dari penulis dalam Teknik Steganografi pada Audio Mp3 menggunakan Metode Parity Coding dan Enkripsi Vigenere Cipher ini adalah sebagai berikut:

1. Penelitian selanjutnya diharapkan dapat di analisa lebih lanjut menggunakan metode lainnya dengan format audio video lainnya seperti AVI, MPG, DAT, MOV, RM, RMVB.
2. Perlu dilakukan Pengembangan untuk meningkatan kapasitas penyisipan dalam arti penyisipan dapat dilakukan selain frame homogen dari berkas Mp3.

DAFTAR PUSTAKA

Ariyus, Dony, 2006, Kriptografi Keamanan Data danKomunikasi, Graha Ilmu, Yogyakarta.

Ariyus, Dony, 2008, Pengantar Ilmu Kriptografi, ANDI offset, Yogyakarta.

Cahyadi, Tri, 2012, Implementasi Steganografi LSB dengan Enkripsi Vigenere Cipher pada Citra JPEG, Jurnal Transient, Vol. 1, No. 4.

Faruq, Umar Al, 2015, Rancang Bangun Aplikasi Rekam Medis Poliklinik Universitas Trilogi, Jurnal Informatika, Vol. 9, No. 1.

Heriyanto, Yunahar, 2018, Perancangan Sistem Informasi Rental Mobil Berbasis Web pada PT. APM RENT CAR, Jurnal Intra-Tech, Vol. 2, No. 2.

Hidayat, Akik, 2009, Kriptografi dan Steganografi Menggunakan Algoritma Vigene dan TEA (Tiny Enkripsion Algorithm), Repositoy UNPAD, Indonesia.

Husodo, Ario Yudo, 2010, Penerapan Metode Enkripsi Vigenere Cipher dalam Pengamanan Transaksi Mobile Banking.

Kusuma, Indra Jaya, 2017, Analisis Teknik Steganografi pada Audio MP3 Menggunakan Metode Parity Coding dan Enkripsi Cipher Transposition, Jurnal STMIK Bina Mulia Palu, Vol. 3, No. 2.

Sallaby, Achmad Fikri, 2015, Aplikasi Widget Berbasis JAVA, Jurnal Media Infotama, Vol. 11 No. 2.

Suendri, 2018, Implementasi Diagram UML (Unified Modelling Language) pada Perancangan Sistem Informasi Remunerasi Dosen dengan Database Oracle, Jurnal Ilmu Komputer dan Informatika, Vol. 03, No. 01.

Tarigan, Andreas Nicolas, 2014, Pembuatan Aplikasi Penyisipan Pesan Pada File MP3 Menggunakan Metode Parity Coding dan Enkripsi Caesar Cipher, Jurnal Pelita Informatika Budi Darma, Vol. VII, No. 2.

Wijaya, Ermadi Satriya, dan Yudi Prayudi, 2004, Konsep Hidden Message Menggunakan Teknik Steganografi Dynamic Cell Spreading, Media Informatika, Vol. 2 No. 1.