PENYEMBUNYIAN DATA EXCEL KEDALAM GAMBAR DENGAN ALGORITMA LSB DAN VERNAM CIPHER

Disusun dan Diajukan Untuk Memenuhi Persyaratan Ujian Akhir Memperoleh
Gelar Sarjana Komputer Pada Fakultas Sains dan Teknologi
Universitas Pembangunan Pancabudi
Medan

SKRIPSI

OLEH

NAMA	**: NAZLIKA PERMATA NUSANTARI RANGKUTI**
NPM	**: 1724370690**
PROGRAM STUDI	**: SISTEM KOMPUTER**

FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS PEMBANGUNAN PANCABUDI
MEDAN
2019

ABSTRAK

NAZLIKA PERMATA NUSANTARI RANGKUTI
Penyembunyian Data Excel kedalam Gambar dengan Algoritma LSB dan Vernam Cipher
2019

Perkembangan teknologi informasi saat ini, semakin memudahkan para pelaku kejahatan komputer, dengan menyalah gunakan teknologi tersebut untuk mendukung kegiatannya, dimana aktivitas mereka sangat mengganggu privasi seseorang. di skripsi ini penyisipan pesan teks dengan motode least Significant Bit. 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 memilih menggunakan kriptografi dengan metode Least Significant Bit, 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 : *Least Significant Bit, Enkripsi, Dekripsi*

KATA PENGANTAR

Puji syukur penulis sampaikan kehadirat Allah SWT, Tuhan Yang Maha Kuasa, atas segala rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan penelitian dan skripsi yang berjudul "**PENYEMBUNYIAN DATA EXCEL KEDALAM GAMBAR DENGAN ALGORITMA LSB DAN VERNAM CIPHER"**.

Terima kasih penulis sampaikan kepada semua pihak yang telah membantu dalam menyelesaikan skripsi ini baik secara langsung maupun tidak langsung. Dalam kesempatan ini penulis ingin mengucapkan terima kasih yang sebesar-besarnya kepada :

1. Yang teristimewa untuk Papa dan Mama tercinta, Bapak Nasry Rizal Rangkuti dan Ibu Hj. Lina Dekriwati Lubis yang telah banyak memberikan doa dan semangat dalam penyusunan skripsi ini, terima kasih atas semua dukungan moral maupun materil sehingga penulis dapat menyelesaikan skripsi ini.
2. Bapak Dr. H. Muhammad Isa Indrawan, S.E., M.M, selaku Rektor Universitas Pembangunan Panca Budi Medan.
3. Ibu Sri Indhira, S.T., M.Sc, selaku Dekan Fakultas Sain dan Teknologi Universitas Pembangunan Panca Budi Medan.
4. Bapak Eko Hariyanto, S.Kom., M.Kom, selaku Ketua Program Studi Fakultas Ilmu Komputer Universitas Pembangunan Panca Budi Medan.
5. Bapak Andysah Putera Utama Siahaan, S.Kom., M.Kom., Ph.D selaku Dosen Pembimbing I yang telah memberi arahan dan masukan bagaimana cara penyusunan, penulisan dan pembelajaran terhadap hal – hal yang sedang dan akan dihadapi tentang skripsi dengan sangat sabar dan baik.
6. Bapak Dr. Muhammad Iqbal, S.Kom., M.Kom, selaku Dosen Pembimbing II yang telah mengajarkan banyak ilmu dan tata cara penulisan skripsi yang baik dan benar.
7. Seluruh teman – teman seperjuangan di Program Studi Sistem Komputer Universitas Pembangunan Panca Budi.
8. Semua pihak yang banyak membantu penulis yang tidak dapat penulis sebutkan satu per satu.

Dalam penyusunan skripsi ini penulis menyadari bahwa masih terdapat kekurangan, oleh karena itu penulis mengharapkan kritik dan saran yang bersifat membangun untuk kesempurnaan skripsi ini agar lebih bermanfaat bagi penulis dan bagi kita semua.

Medan, Oktober 2019
Penulis,

Nazlika Permata Nusantari Rangkuti
NPM. 1724370690

DAFTAR ISI

Halaman

DAFTAR GAMBAR

Halaman

DAFTAR LAMPIRAN

DAFTAR TABEL

Halaman

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi informasi semakin memudahkan penggunanya dalam berkomunikasi melalui bermacam-macam media. Komunikasi yang melibatkan pengiriman dan penerimaan pesan dengan memanfaatkan kemajuan teknologi informasi rentan terhadap pelaku kejahatan komputer yang memanfatkan celah keamanan untuk mendeteksi dan memanipulasi pesan.

Keamanan dan kerahasiaan menjadi aspek yang sangat penting bagi pengguna teknologi informasi. Untuk menghindari pesan yang dikirimkan jatuh pada pihak-pihak yang tidak berkepentingan dan terjadi penyalahgunaan terhadap pesan, maka dilakukan enkripsi terhadap pesan asli dan penyisipan pesan ke dalam suatu media dengan menerapkan ilmu steganografi.

Untuk meningkatkan keamanan digunakan steganografi, dimana suatu sistem steganografi sedemikian rupa menyembunyikan isi suatu informasi di dalam suatu media yang tidak dapat di duga oleh orang biasa sehingga tidak membangunkan suatu kecurigaan kepada orang yang melihatnya. Media untuk menyembunyikan informasi adalah Format *image* diantaranya bitmap (bmp) , gif, pcx, dan jpeg. Format *audio* antara lain wav, mp3, voc. Format lain misalkan teks file, doc, html dan pdf. Tujuan dari steganografi adalah merahasiakan atau menyembunyikan keberadaan dari sebuah informasi. Terutama file yang sering disalahgunakan pihak yang tidak berkepentingan adalah file Ms. Excel. Dimana

file ini biasanya berisi informasi penting suatu organisasi ataupun perusahaan seperti infromasi tentang keuangan dan lain sebagainya. Maka dari itu untuk membantu masyarakat, organisasi, ataupun suatu lembaga dalam hal pengamanan data berupa file Ms. Excel ke dalam sebuah media berupa gambar, penulis ingin membuat skripsi dengan judul **"Penyembunyian Data Excel Ke Dalam Gambar Menggunakan Algoritma LSB (Least Significant Bit) dan Vernam Chiper"**

1.2 Rumusan Masalah

Berdasarkan latar belakang diatas, adapun rumusan masalah yang akan dibahas penulis adalah

1. Bagaimana merancang sebuah keamanan data menggunakan file *Microsoft Excel* dan Algoritma *LSB (Least Significant Bit)* menggunakan vernam chiper?
2. Bagaimana menerapkan metode algoritma *LSB (Least Significant Bit)* dan *vernam chiper* dalam proses keamanan data pada *Microsoft Excel ?*

1.3 Batasan Masalah

Berdasarkan perumusan masalah diatas maka penulis melakukan pembatasan masalah yang akan dibahas sebagai berikut:

1. Implementasi enkripsi menggunakan kriptografi klasik yaitu, vigenere chiper dan dekripsi menggunakan file Microsoft Excel dengan algoritma LSB (Least Significant Bit) dan vernam chiper.

2. Program yang dibahas menggunakan pemrograman Visual Basic.Net 2010.

1.4 Tujuan Penelitian

Adapun tujuan dari penelitian dengan menggunakan algoritma *LSB (Least Significant Bit)* dan vernam chiper ini yang ingin dicapai adalah sebagai berikut:

1. Merancang sistem aplikasi keamanan data dengan algoritma algoritma LSB (Least Significant Bit) dan vernam chiper pada file Microsoft Excel.
2. Memperkuat keamanan data sebuah file yang bersifat rahasia terutama data yang menggunakan file Microsoft Excel.

1.5 Manfaat Penelitian

Adapun manfaat dalam penelitian ini yang diperoleh dari penerapan dengan algoritma *LSB (Least Significant Bit)* adalah sebagai berikut:

1. Kerahasiaan data yang dikirim dan diterima lebih aman.
2. Sebagai media pembelajaran dalam bidang keamanan informasi dan keamanan data.

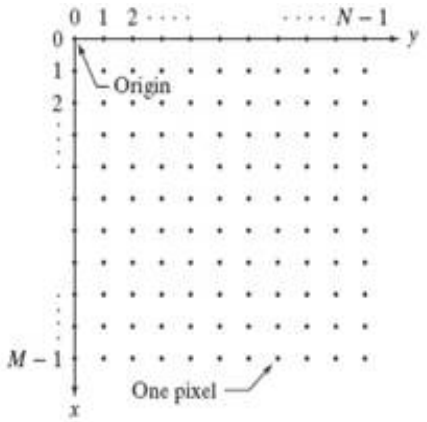
BAB II

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 komplek yang direpresentasikan dengan deretan bit tertentu (Putra, 2010). 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.

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 : Kadir, 2010)

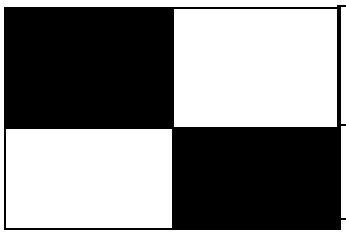
Dengan sistem koordinat yang mengikuti asas pemindaian pada layar TV standar itu, sebuah piksel mempunyai koordinat berupa (x, y) dalam hal ini:

1. x menyatakan posisi kolom;
2. y menyatakan posisi baris;
3. 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:

2.2 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.



Gambar 2.2. Contoh Citra Biner Berukuran 2x2 *Pixel*

(Sumber : Kadir, 2010)

2.3 Citra Warna

Setiap piksel pada citra warna memiliki warna yang merupakankombinasi 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 . 28 . 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.

2.4 Pengolahan Citra Digital

2.4.1 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 output-nya adalah citra hasil pengolahan (Putra, Pengolahan Citra Digital, 2010).

2.4.2 Tujuan Pengolahan Citra Digital

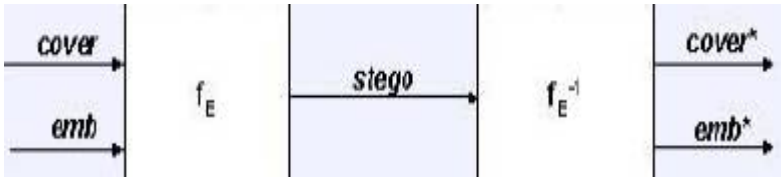
Pengolahan citra digital banyak dimanfaatkan oleh berbagai bidang mulai dari keamanan, kesehatan, pendidikan dan bidang – bidang yang lain. Berikut beberapa tujuan dari kegiatan pengolahan citra digital.

1. Memperbaiki kualitas gambar dilihat dari aspek *radiometric* (peningkatan kontras, tranformasi 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(Sutoyo, 2009).

2.5 Steganografi

Steganografi adalah ilmu dan seni dari komunikasi yang tidak terlihat (Morkel, Eloff, & Olivier, 2005). Steganografi merupakan kata yang diturunkan dari kata-kata yunani yaitu "*stegos"* yang berarti "menutupi" dan "*grafia"* yang berarti menulis yang mana jika didefinisikan dapat dengan "tulisan yang ditutupi". Steganografi berbeda dari kriptografi dimana kriptografi bertujuan pada menjaga konten atau informasi dari pesan tetap rahasia sedangkan steganografi bertujuan untuk menjaga keberadaan pesan tetap rahasia.

Pesan asli disembunyikan pada sebuah media pembawa yang mana perubahan yang terjadi pada media pembawa tidak terlihat oleh orang lain (Kumar & Pooja, 2010). Kelebihan dari steganografi salah satunya adalah dimana pesan ditransmisikan atau dikirim tanpa diketahui oleh pihak lain yang mana bagi pihak lain yang terlihat adalah media pembawanya saja.

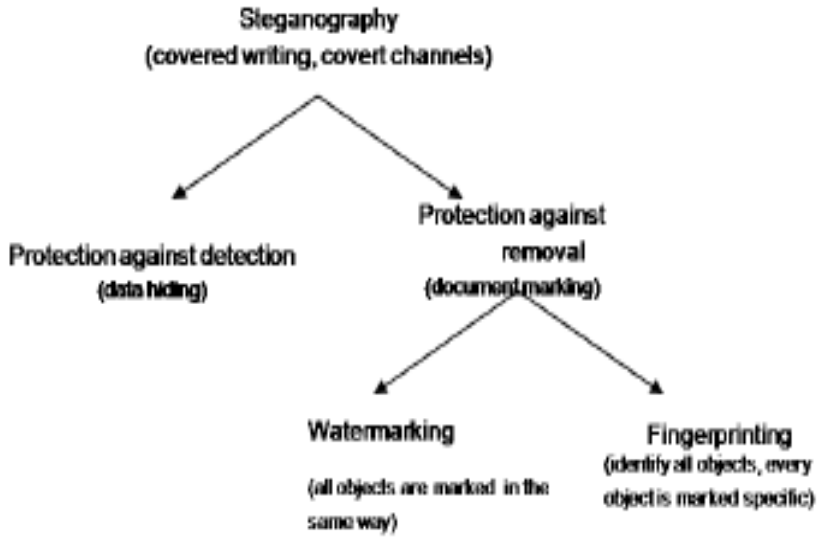


Gambar 2.3. Ilustrasi Sistem Steganografi.

(Sumber : Kumar & Pooja, 2010)

Penggunaan steganografi adalah sebagai berikut :

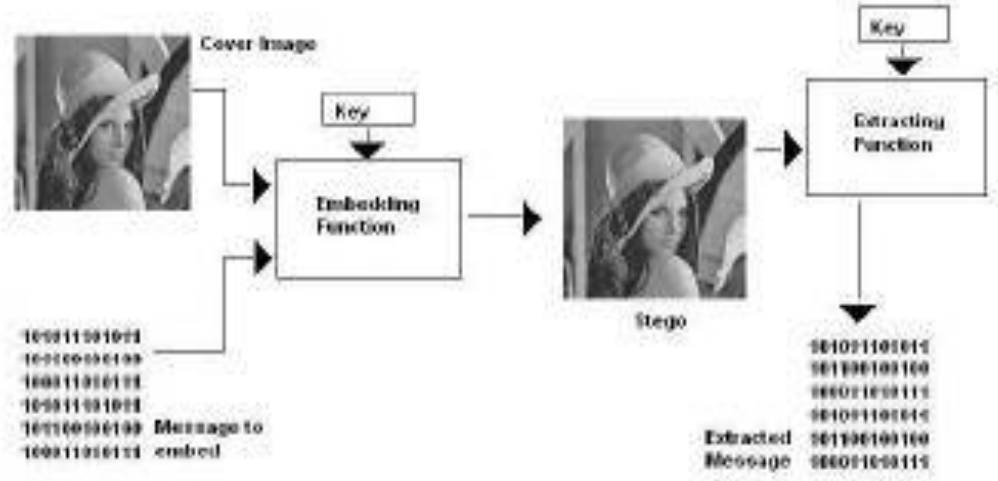
1. Steganografi dapat menjadi solusi yang mana memungkinkan untuk mengirim berita atau informasi dicegah oleh sensor atau khawatir terhadap pesan dibajak oleh pihak lain.
2. Steganografi juga dapat digunakan untuk menyimpan pada suatu lokasi seperti media digital lain.
3. Steganografi juga dapat digunakan sebagai watermarking pada media yang ingin dilindungi hak ciptanya.



Gambar 2.4. Tipe dari Steganografi.

(Sumber : Kumar & Pooja, 2010)

Semua pendekatan yang ada pada bidang steganografi memiliki sebuah kesamaan yaitu menyembunyikan pesan rahasia pada objek fisik yang dikirimkan. Pada gambar diatas dapat dilihat proses dari steganografi dimana citra pembawa diteruskan kedalam fungsi penanaman yang kemudian akan menghasilkan citra yang telah mengandung pesan rahasia. Proses steganografi juga biasanya dapat menggunakan kunci untuk meningkatkan keamanan pada pesan yang disembunyikan, yang mana proses steganografi akan dilengkapi dengan proses kriptografi sebagai proses tambahan.



Gambar 2.5. Prosedur Steganografi.

(*Sumber : Kumar & Pooja, 2010*)

2.6 Steganografi LSB dan Vernam Cipher

LSB atau *Least Significant Bit* merupakan teknik yang umum digunakan pada bidang steganografi. Metode LSB bekerja dengan mengganti bit pada posisi *least significant* dengan bit dari informasi yang akan ditanam (Juneja, Sandhu, & Walia, 2009). Berikut ilustrasi dari proses penanaman informasi menggunakan steganografi LSB pada media citra digital.

Proses *Embedding* :

Piksel : 00100111 11101001 11001000)

(00100111 11001000 11101001)

(11001000 00100111 11101001)

Karakter : A -> 65 -> 01000001

Hasil : (00100110 11101001 11001000)

(00100110 11001000 11101000)

(11001000 00100111 11101001)

Proses *embedding* atau penanaman dilakukan dengan cara mengganti bit LSB pada citra dengan bit dari karakter informasi. Bit yang digaris bawah seperti yang terlihat pada proses *embedding* diatas merupakan bit pengganti yan diperoleh dari karakter informasi. Proses ekstraksi dilakukan dengan mengambil bit LSB dari tiap piksel dan kemudian merangkainya kembali menjadi karakter informasi.

Proses *Extracting* :

Hasil : (00100110 11101001 11001000)

(00100110 11001000 11101000)

(11001000 00100111 11101001)

Ekstraksi Bit : 0 1 0 0 0 0 1 1

Desimal : 65

Karakter : A

Vernam cipher merupakan algoritma kriptografi yang ditemukan oleh Mayor J. Maugborne dan G. Vernam. Algoritma ini merupakan algoritma berjenis symmetric key yang artinya bahwa kunci yang digunakan untuk melakukan enkripsi dan dekripsi merupakan kunci yang sama. Dalam proses enkripsi, algoritma ini menggunakan cara stream cipher dimana cipher berasal dari hasil XOR antara bit plaintext dan bit key. Algoritma *Vernam cipher* diadopsi dari one-time pad cipher, dimana dalam hal ini karakter diganti dengan bit (0atau1). Dengan kata lain, Vernam Cipher merupakan versi lain dari one-time pad cipher (Wicaksono, 2011). Vernam cipher adalah jenis algoritma enkripsi simetri. Vernam cipher dapat dibuat sangat cepat sekali, jauh lebih cepat dibandingkan dengan algoritma block cipher yang manapun. Algoritma block cipher secara umum digunakan untuk unit plaintext yang besar sedangkan stream cipher digunakan untuk blok data yang lebih kecil, biasanya ukuran bit (Stinson, 1995). Proses enkripsi terhadap plaintext tertentu dengan algoritma block cipher akan menghasilkan ciphertext yang sama jika kunci yang sama digunakan. Dengan stream cipher, transformasi dari unit plaintext yang lebih kecil ini berbeda antara satu dengan lainnya, tergantung pada kapan unit tersebut ditemukan selama proses enkripsi. Satu vernam cipher menghasilkan apa yang disebut suatu keystream (suatu barisan bit yang digunakan sebagai kunci). Proses enkripsi dicapai dengan menggabungkan keystream dengan plaintext biasanya dengan operasi bitwise XOR (Kromodimoeljo, 2009). Pembentukan keystream dapat dibuat independen terhadap plaintext dan ciphertext, menghasilkan synchronous stream cipher, atau dapat dibuat tergantung pada data dan enkripsinya, dalam hal

mana stream cipher disebut sebagai self-synchronizing. Kebanyakan bentuk stream cipher adalah synchronous stream cipher.

2.7 Microsoft Visual Basic 2010

Pada perancangan perangkat lunak ini, penulis mempergunakan Microsoft visual basic 2010 sebagai bahasa pemrogramannya. Visual Basic 2010 merupakan salah satu bagian dari produk pemrograman terbaru yang dikeluarkan oleh Microsoft yaitu Microsoft Visual Basic 2010. Microsoft Visual Basic 2010 berupa bahasa pemrograman, yang menghasilkan aplikasi-aplikasi pada Windows yang berbasis grafis (GUI-*Grafical user interface*). Visual Basic 2010 ini menambahkan perbaikan-perbaikan fitur dan fitur baru yang lebih lengkap dibandingkan versi Visual Studio pendahulunya, yaitu Microsoft Visual Studio 2008. (Prabawati, 2010).

Microsoft Visual Studio adalah sebuah perangkat lunak lengkap (*suite*) yang dapat digunakan untuk melakukan pengembangan aplikasi, baik itu aplikasi bisnis, aplikasi personal, ataupun komponen aplikasinya, dalam bentuk aplikasi console, aplikasi Windows, ataupun aplikasi Web. Visual Studio 2010 mencakup kompiler, SDK, *Integrated Development Environment* (IDE), *Visual Web Developer*, dan dokumentasi (umumnya berupa MSDN Library). Kompiler yang dimasukkan ke dalam paket Visual Studio antara lain *Visual C++, Visual C#, Visual Basic, Visual Basic .NET, Visual InterDev, Visual J++, Visual J#, Visual FoxPro, dan Visual SourceSafe*.

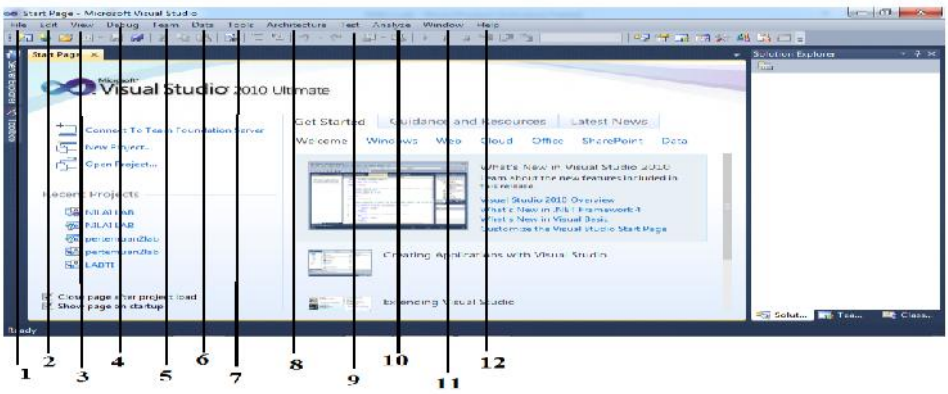
Microsoft Visual Studio dapat digunakan untuk mengembangkan aplikasi dalam *native code* (dalam bentuk bahasa mesin yang berjalan di atas Windows) ataupun

managed code dalam bentuk *Microsoft Intermediate Language* di atas .NET *Framework*). Selain itu, Visual Studio juga dapat digunakan untuk mengembangkan aplikasi Silverlight, aplikasi Windows Mobile (yang berjalan di atas .NET *Compact Framework*).

Visual Basic dapat berupa *software* yang dirancang untuk membuat aplikasi pada komputer. Awalnya *software* ini memang diarahkan untuk membuat aplikasi berbasis *desktop*, tetapi pada perkembangannya banyak digunakan untuk membuat aplikasi yang bukan berbasis internet (*offline*). Penggunaan *microsoft visual basic* untuk pembuatan aplikasi tidaklah sulit, *tool-tool* yang tersedia cukup mudah digunakan, beberapa *template* dan komponen juga sudah disediakan dan siap digunakan.

2.7.1 User Interface Microsoft Visual Basic 2010

User interface dari Microsoft Visual Basic 2010 diperlihatkan pada gambar 2.6



Gambar 2.6 user interface dari Microsoft visual Basic

Fungsi Menu User Interface Microsoft Visual Basic diperlihatkan pada tabel 2.1

Tabel 2.1 Fungsi Menu User Interface Microsoft Visual Basic

NO	NAMA	KETERANGAN
1.	*File*	Kelompok perintah yang berfungsi mengatur *file* seperti *new, save, open, remove, print* dan lain-lain.
2.	*Edit*	Kelompok perintah untuk pengeditan baik objek, komponen maupun kode pada editor.
3.	*View*	Perintah-perintah untuk mengaktifkan bagian-bagian ide *visual basic*.
4.	*Debug*	Perintah dalam pencarian kesalahan program.
5.	*Team*	Perintah untuk *connect* ke *server* tertentu.
6.	*Data*	Perintah *connect* ke *sql server*.
7.	*Tools*	Sebagai penyedia perlengkapan tambahan yang diperlukan dalam penyusunan program.
9.	*Test*	Perintah untuk *run* program.
10.	*Analyze*	Perintah untuk menganalisa program.
11.	*Window*	Perintah mengatur desain form.
12.	*Help*	Menyediakan informasi untuk menolong pemakai.

2.7.2 Fungsi Komponen – Komponen Microsoft Visual Basic 2010

Komponen merupakan bagian dari perlengkapan suatu aplikasi yang mempunyai spesifikasi properti sendiri. Komponen-komponen pada Microsoft Visual Studio 2010 adalah:

1. *Menu*

   Menu adalah bagian dari IDE yang terdiri dari perintah-perintah untuk mengatur IDE, mengembangkan, memelihara dan mengeksekusi program. Di dalam menu, perintah-perintah dikelompokkan ke dalam beberapa bagian sesuai jenis perintah menu pada *Visual basic*.

2. *Toolbar*

   Toolbar fungsinya sama seperti fungsi dari menu, hanya saja pada *toolbar* pilihan-pilihan berbentuk *icon*. Untuk memilih suatu proses yang akan dilakukan, tinggal mengklik *icon* yang sesuai dengan proses yang diinginkan. *Icon-icon* pada *toolbar* adalah pilihan-pilihan pada menu yang sering digunakan dalam membuat program aplikasi. Dengan adanya *toolbar*, untuk memilih proses yang sering dilakukan tanpa harus memilihnya pada menu.

3. *Toolbox*

   Toolbox adalah tempat kontrol dan komponen-komponen diletakkan. Kontrol dan komponen yang terdapat pada *toolbox* dipakai dalam pembuatan program aplikasi. Untuk membuat objek kontrol dan komponen pada *form* program aplikasi diambil dari kontrol-kontrol yang ada pada *toolbox*.

4. *Server Explorer*

   Server Explorer adalah bagian tempat untuk mengatur hal-hal yang berhubungan dengan *server* dan *database.*

5. *Solution Explorer*

   Solution Explorer memberikan tampilan daftar *file-file project* yang sedang dibuat sehingga dapat diakses langsung. Pada *windows solution explorer* terdapat beberapa tombol pada *toolbar* dan *tree* yang berisi daftar *file-file* yang digunakan dalam *project.*

6. *Properties Window*

   Properties windows adalah tempat untuk daftar properti setiap objek kontrol dan komponen. *Properties window* juga dipakai untuk mengatur properti objek kontrol dan komponen yang dipakai. Dengan *properties window*, dapat mengubah properti yang nantinya akan dipakai sebagai *default* objek kontrol dan komponen pada waktu pertama kali program dieksekusi.

7. *Form*

   Form adalah tempat membuat tampilan (*user interface*) untuk program aplikasi. Pada *form user* dapat meletakkan atau menambahkan objek kontrol maupun komponen.

8. Kode Editor

   Kode editor adalah tempat meletakkan atau menuliskan kode program dari program aplikasi. Pada kode editor juga terdapat bagian objek dan *event* dari *control*.

2.8 *Flowchart*

Flowchart merupakan gambar atau bagan yang memperlihatkan urutan dan hubungan antar proses beserta instruksinya (Blauch, 2012). Gambaran ini dinyatakan dengan simbol. Dengan demikian setiap simbol menggambarkan proses tertentu. Sedangkan hubungan antar proses digambarkan dengan garis penghubung. *Flowchart* ini merupakan langkah awal pembuatan program.

Dengan adanya *flowchart* urutan proses kegiatan menjadi lebih jelas. Jika ada penambahan proses maka dapat dilakukan lebih mudah. Setelah *flowchart* selesai disusun, selanjutnya pemrogram (programmer) menerjemahkannya ke bentuk program dan bahasa pemrograman.

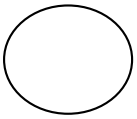
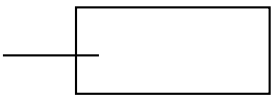
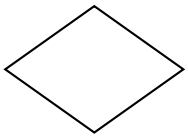
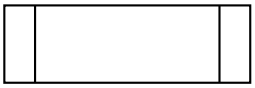
1. *Flowchart* Sistem *(System Flowchart)*

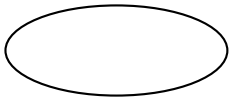
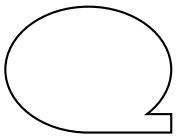
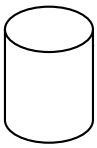
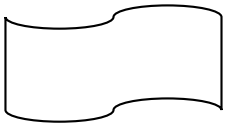
   Sistem *Flowchart* merupakan bagian yang menunjukkan alur kerja atau apa yang sedang dikerjakan di dalam sistem secara keseluruhan dan menjelaskan urutan dari prosedur-prosedur yang ada di dalam sistem. Dengan kata lain, *flowchart* ini merupakan deskripsi secara frafik dari urutan prosedur-prosedur yang terkombinasi yang membentuk suatu sistem.

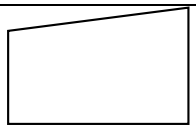
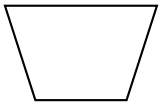
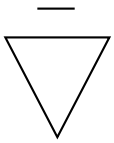
   Flowchart Sistem terdiri dari data yang mengalir melalui system dan proses yang mentransformasikan data itu. Data dan proses dalam *flowchart* sistem dapat digambarkan secara *online* (dihubungkan langsung dengan computer) atau offline(tidak dihubungkan langsung denan computer, misalnya mesin tik, cash register atau kalkulator).

   Simbol-simbol yang digunakan dalam system *flowchart* antara lain :

Tabel 2.2. Simbol-simbol *flowchart*.

SIMBOL	NAMA SIMBOL / ARTI
	INPUT / OUTPUT Mempresentasikan input data atau output data yang diproses atau informasi
	PROSES Mempresentasikan operasi
	PENGHUBUNG Keluar atau masuk dari bagian lain *flowchart* khususnya halaman yang sama
	ANAK PANAH Mempresentasikan alur kerja
	PENJELASAN Digunakan untuk komentar tambahan
	KEPUTUSAN Keputusan dalam program
	PREDEFINED PROCESS Rincian operasi berada di tempat lain.
	PREPARATION Pemberian harga awal

	TERMILAN POINTS Awal / akhir *flowchart*
	PUNCHED CARD Input / output yang menggunakan kartu berulang
	DOKUMEN Input / output dalam format yang dicetak
	MAGNETIC TAPE Input / output yang menggunakan pita magnetic
	MAGNETIC DISK Input / Output yang menggunakan disk magnetic
	ON-LINA STORAGE Input / output yang menggunakan penyimpanan akses langsung
	PUNCHED TAPE Input / output yang menggunakan pita kertas berlubang

	MANUAL INPUT Input yang dimasukkan secara manual dari keyboard
	DISPLAY Output yang ditampilkan pada terminal
	MANUAL OPERATION Operasi manual
	OFF – LINE STORAGE Penyimpanan yang tidak dapat diakses oleh komputer secara langsung
	COMMUNICATION LINK Transmisi data melalui channel komunikasi, Seperti telepon

(Sumber : Blauch, 2012)

2.9 UML

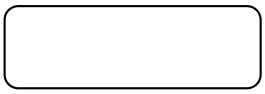
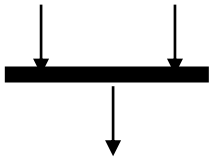
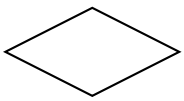
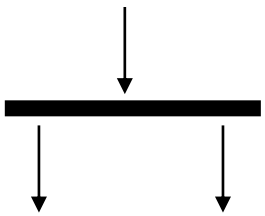
UML *(Unified Modeling Laguange*) adalah sebuah bahasa yang sudah menjadi standar dalam industry untuk merancang, menspesifikasi dan mendokumentasi sistem perangkat lunak (Seidl & Huemer, 2012). Adapun tujuan dari UML adalah:

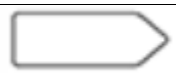
1. Merancang perangkat lunak.
2. Sarana komunikasi antara perangkat lunak dengan proses bisnis.
3. Menjabarkan sistem secara rinci untuk analisa dan mencari apa yang diperlukan sistem.
4. Mendokumentasi sistem yang ada, proses-proses dan organisasinya.

2.10 *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 (Alami & Ferati, 2016). 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 2.3. Notasi *Activity* Diagram

Simbol	Keterangan
	Titik Awal
	Titik Akhir
	*Activity*
	*Connector*
	*Join*
	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: (Alami & Ferati, 2016)

2.11 *Sequence Diagram*

Sequence Diagram mendeskripsikan skenario (dapat mengacu pada expanded use case yang telah dibuat) dalam bentuk diagram (Kharisma, 2014). 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 diatasnya diberikan label pesan.

Tabel 2.4. Simbol Sequence Diagram

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

Sumber: (Kharisma, 2014)

2.12 *Use Case Diagram*

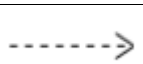
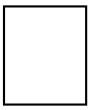
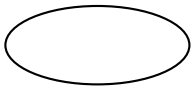
Use case diagram digunakan untuk menspesifikasikan fungsionalitas dari sistem (Eccles, 2015). *Use case* merupakan sebuah pekerjaan tertentu, misalnya login ke sistem, meng-*create* sebuah daftar belanja, dan sebagainya.

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 2.5. Simbol *Use Case* Diagram

Gambar	Nama	Keterangan
	*Actor*	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinterasi dengan *Use Case*.
	*Depedency*	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (*independent)* akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (*Independent)*
	Generalization	Hubungan dimana objek anak(*Descended)* berbagi prilaku dan struktur data dari objek yang diatasnya objek induk.

	*Include*	Menspesifikasikan bahwa use case sumber secara explicit.
	*Extend*	Menspesifikasikan bahwa use case target memperluas prilaku pada use case sumber pada sebuah titik diberikan.
	Assosiation	Apa yang memhubungkan objek satu dengan objek yang lainnya.
	*System*	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
	*Use Case*	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur dari sebuah *actor.*
	*Colaboration*	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan prilaku yang lebih besar dari jumlah dan elemen-elemennya.
	*Note*	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu

		sumber daya komputasi.

Sumber: (Erlita, 2014)

2.13 Metode-Metode Kriptografi

Kriptografi berasal dari bahasa Yunani, "*kryptós*" yang berarti tersembunyi dan "gráphein" yang berarti tulisan. Sehingga kata kriptografi dapat diartikan menjadi "tulisan tersembunyi". kriptografi adalah ilmu matematika yang berhubungan dengan transformasi data agar arti dari data tersebut menjadi sulit untuk dipahami, mencegahnya dari perubahan tanpa izin, atau mencegahnya dari penggunaan yang tidak sah. (Zelvina, 57 : 2012).

2.14 Tujuan kriptografi

Tujuan dari kriptografi yang juga merupakan aspek keamanan informasi adalah sebagai berikut: (Zelvina, 58 : 2012)

1) Kerahasiaan (*confidentiality*) adalah layanan yang digunakan untuk menjaga isi informasi dari semua pihak kecuali pihak yang memiliki otoritas terhadap informasi. Ada beberapa pendekatan untuk menjaga kerahasiaan, dari pengamanan secara fisik hingga penggunaan algoritma matematika yang membuat data tidak dapat dipahami. Istilah lain yang senada dengan *confidentiality* adalah *secrecy* dan *privacy*.
2) Integritas data adalah layanan penjagaan pengubahan data dari pihak yang tidak berwenang. Untuk menjaga integritas data, sistem harus memiliki kemampuan untuk mendeteksi manipulasi pesan oleh pihak-pihak yang tidak berhak, antara lain penyisipan, penghapusan, dan pensubsitusian data lain

kedalam pesan yang sebenarnya. Di dalam kriptografi, layanan ini direalisasikan dengan menggunakan tanda-tangan digital (*digital signature*). Pesan yang telah ditandatangani menyiratkan bahwa pesan yang dikirim adalah asli.

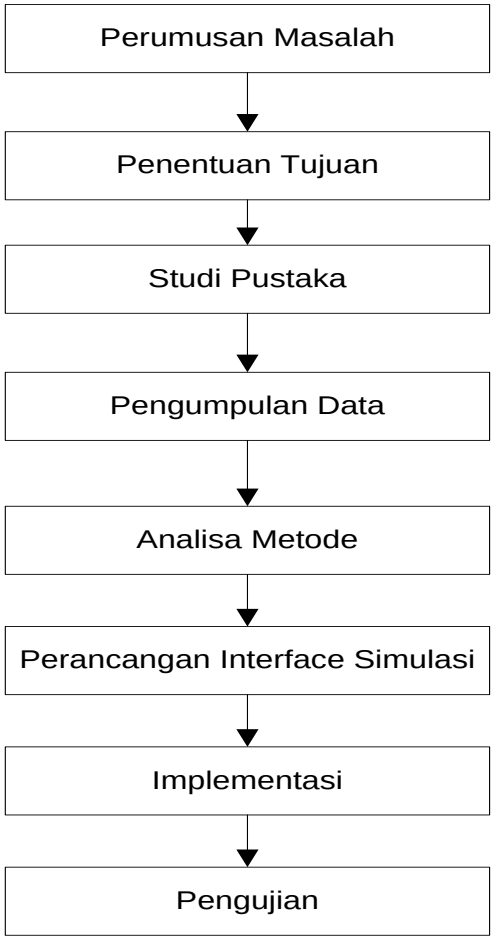
3) Otentikasi adalah layanan yang berhubungan dengan identifikasi, baik mengidentifikasi kebenaran pihak-pihak yang berkomunikasi (*user authentication* atau *entity authentication*) maupun mengidentifikasi kebenaran sumber pesan (*data origin authentication*). Dua pihak yang saling berkomunikasi harus dapat mengotentikasi satu sama lain sehingga ia dapat memastikan sumber pesan. Pesan yang dikirim melalui saluran komunikasi juga harus diotentikasi asalnya. Otentikasi sumber pesan secara implisit juga memberikan kepastian integritas data, sebab jika pesan telah dimodifikasi berarti sumber pesan sudah tidak benar. Oleh karena itu, layanan integritas data selalu dikombinasikan dengan layanan otentikasi sumber pesan. Di dalam kriptografi, layanan ini direalisasikan dengan menggunakan tanda-tangan digital (*digital signature*). Tanda-tangan digital menyatakan sumber pesan.
4) Nirpenyangkalan (*non-repudiation*) adalah layanan untuk mencegah entitas yang berkomunikasi melakukan penyangkalan, yaitu pengirim pesan menyangkal melakukan pengiriman atau penerima pesan menyangkal telah menerima pesan.

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) dan Vernam Chiper adalah sebagai berikut:



Gambar 3.1 Tahapan Penelitian

3.2 Metode Pengumpulan Data

Pengumpulan data adalah pencarian terhadap susuatu kerena ada perhatian dan keinginan terhadap hasil suatu aktivitas. Metode pengumpulan data dalam penulisan ini dibagi menjadi 3, yaitu :

1. Pengamatan (*Observation*)

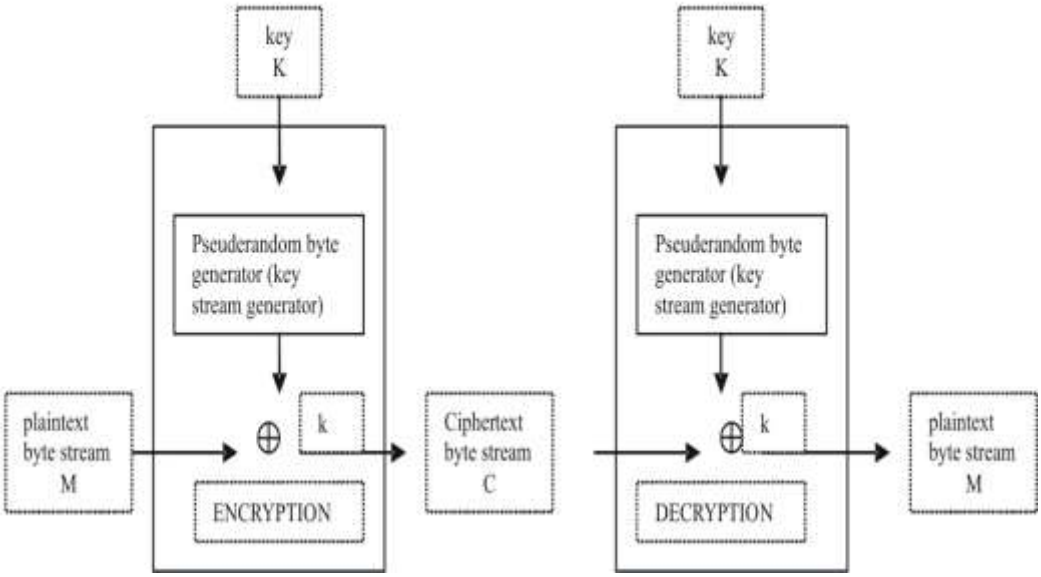
   Penulis melakukan pengamatan langsung pada setiap penggunan aplikasi chating yang sudah ada seperti WA, BBM dan Line untuk mengamati proses keamanan yang sudah dibuat sebelumnya.

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 Permasalah 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* dan Vernam Chiper. 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 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 masukkan yan diperlukan sistem, keluaran yang akan dihasilkan sistem dan proses yang dibutuhkan untuk mengolah masukkan 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 10 sebagai sistem operasi
b. Microsoft Visual Studio 2010

3.7 Analisa Proses Sistem Yang Berjalan

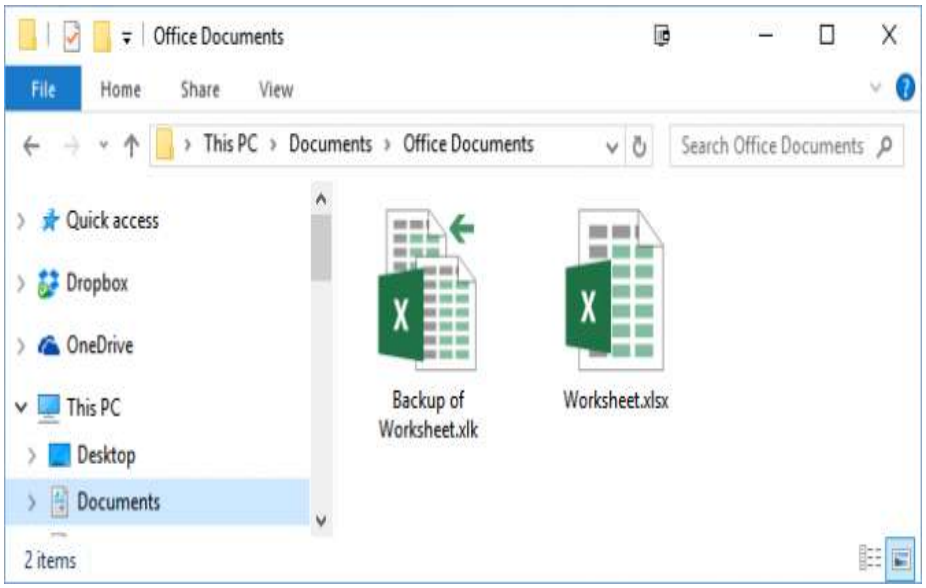
Terdapat 2 (dua) proses utama dalam penyisipan pesan menggunakan metode *Least Significant Bit*, yaitu proses *embedding* dan proses *extraction.* Proses *embedding* adalah proses penyisipan pesan rahasia ke dalam suatu media. Sedangkan proses *extraction* adalah proses pengambilan pesan rahasia dari 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 teks AKU

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

00000001	00010100	00000000	00000001	00010100	00000000	00000001	00010100
00000001	00000000	00010011	00000000	00000000	00010011	00000000	00000000
00010101	00000000	00000000	00010110	00000001	00000000	00011000	00000000
00000000	00011010	00000000	00000001	00010100	00000000	00000000	00010011
00000000	00000000	00010011	00000000	00000000	00010110	00000001	00000000
00010110	00000001	00000000	00010110	00000001	00000010	00010101	00000010
00000000	00010011	00000000	00000001	00010011	00000011	00000000	00010001
00000001	00000000	00010001	00000001	00000000	00010000	00000000	00000000

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

00000**0**	00000**1**	00010**0**	00000**0**	00000**0**	00010**0**	00000**0**	0000000**1**	A
00010**0**	00000**1**	00000**0**	00010**1**	00000**1**	00000**0**	00011**0**	00000001	K
00000**0**	00011**1**	00000**0**	00000**1**	00011**0**	00000**1**	00000**0**	00011001	U
00000000	00000000	00010101	00000000	00000000	00010011	00000000	00000000	-
00010011	00000000	00000000	00010111	00000001	00000000	00010111	00000001	-
00000000	00010111	00000001	00000010	00010101	00000010	00000000	00010011	-
00000000	00000000	00010011	00000011	00000000	00010001	00000001	00000000	-
00010001	00000001	00000000	00010001	00000000	00000000	00010001	00000000	-

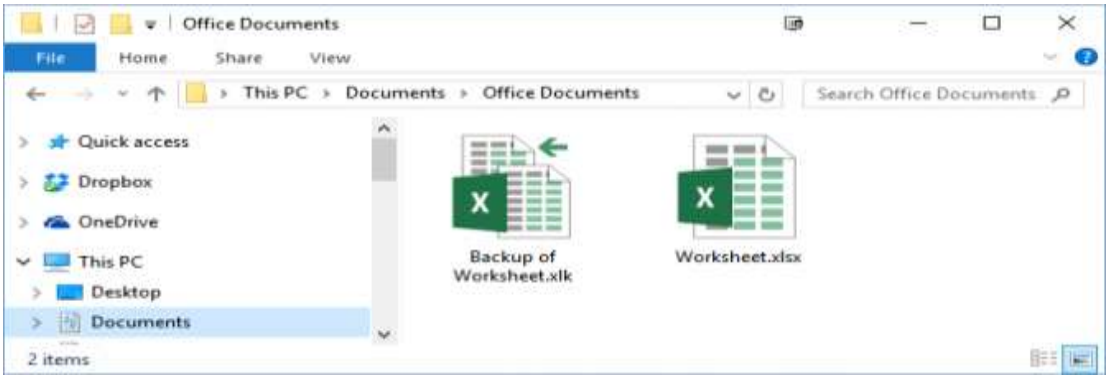
Terlihat pada tiap akhir dari biner Ms. Excel telah tersisipi oleh pesan rahasia yang ditandai dengan huruf *Bold* (cetak tebal). Langkah selanjutnya adalah matriks tersebut akan dipetakan kembali dalam bentuk Ms. Excel dan Ms. Excel ini disebut *stego file.*

2. Proses *Extraction*

Proses *extraction* atau pengambilan *text* dari media penampung menggunakan metode *Least Significant Bit* adalah sebag ai 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 7.

Tabel 3.5. Tabel biner Ms. Excel yang berisi pesan rahasia

0000000**0**	0000000**1**	0001001**0**	0000000**0**	0000000**0**	0001001**0**	0000000**0**	0000000**1**	A
0001010**0**	0000000**1**	0000000**0**	0001011**0**	0000000**1**	0000000**0**	0001100**1**	0000000 1	K
0000000**0**	0001101**1**	0000000**0**	0000000**1**	0001100**0**	0000000**1**	0000000**0**	0001100**1**	U
0000000 0	0000000 0	0001010 1	0000000 0	0000000 0	0001001 1	0000000 0	0000000 0	-
0001001 1	0000000 0	0000000 0	0001011 1	0000000 1	0000000 0	0001011 1	0000000 1	-
0000000 0	0001011 1	0000000 1	0000001 0	0001010 1	0000001 0	0000000 0	0001001 1	-
0000000 0	0000000 0	0001001 1	0000001 1	0000000 0	0001000 1	0000000 1	0000000 0	-
0001000 1	0000000 1	0000000 0	0001000 1	0000000 0	0000000 0	0001000 1	0000000 0	-

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

Tabel 3.6. Tabel biner pesan rahasia yang disisipkan

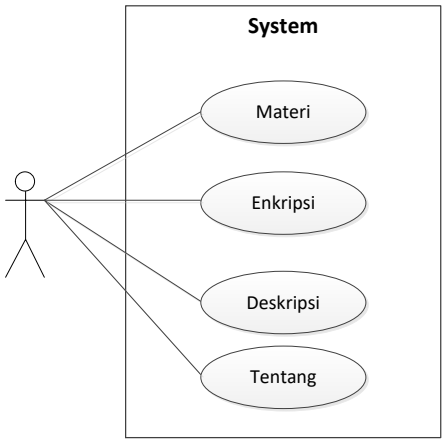
0000000**0**	0000000**1**	0001001**0**	0000000**0**	0000000**0**	0001001**0**	0000000**0**	0000000**1**	A
0001010**0**	0000000**1**	0000000**0**	0001011**0**	0000000**1**	0000000**0**	0001100**1**	0000000 1	K
0000000**0**	0001101**1**	0000000**0**	0000000**1**	0001100**0**	0000000**1**	0000000**0**	0001100**1**	U

3.8 Perancangan Berorientasi Objek

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.



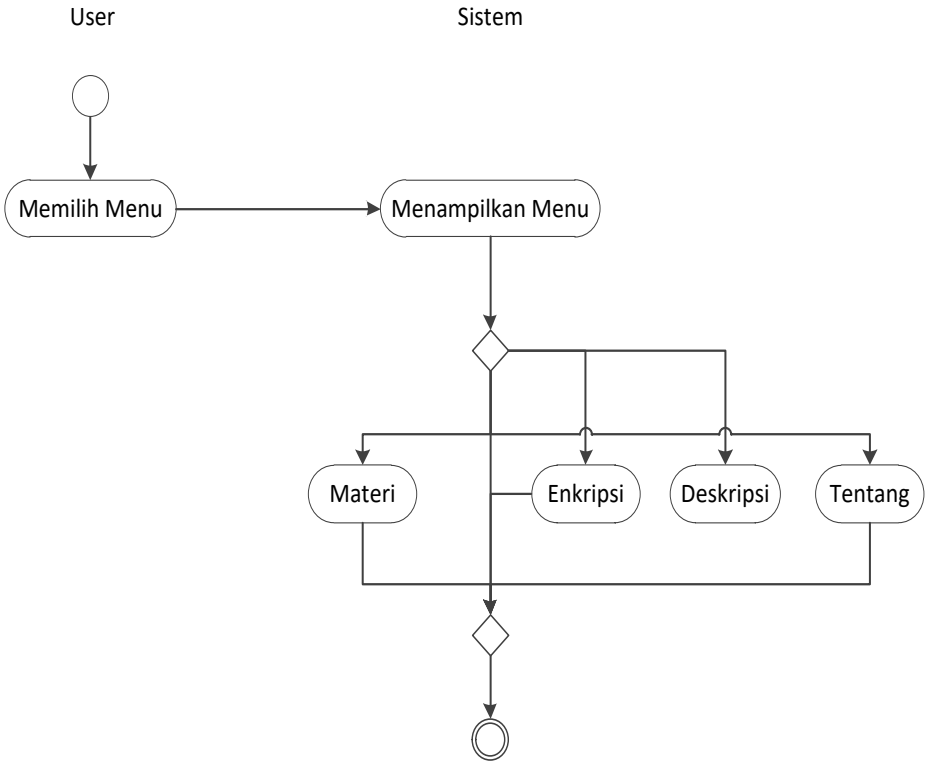
Gambar 3.5. Use Case Diagram

Keterangan :

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

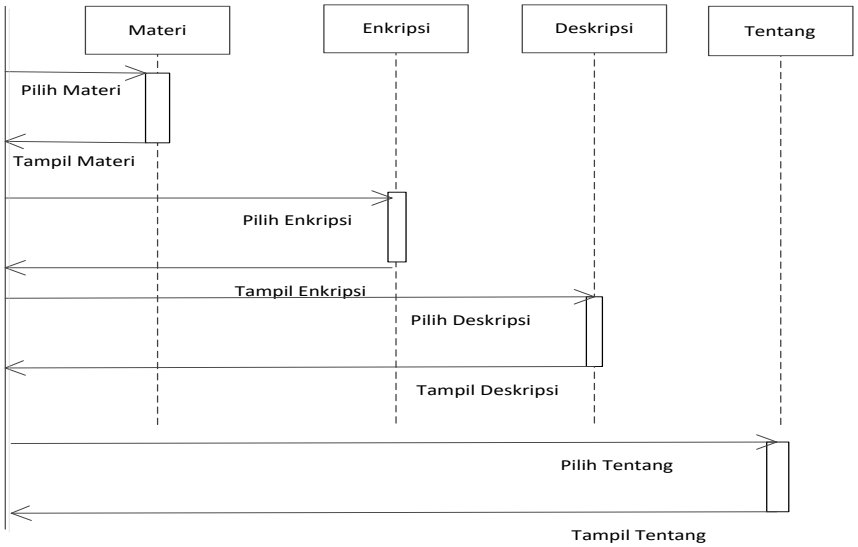
2. Activity Diagram

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

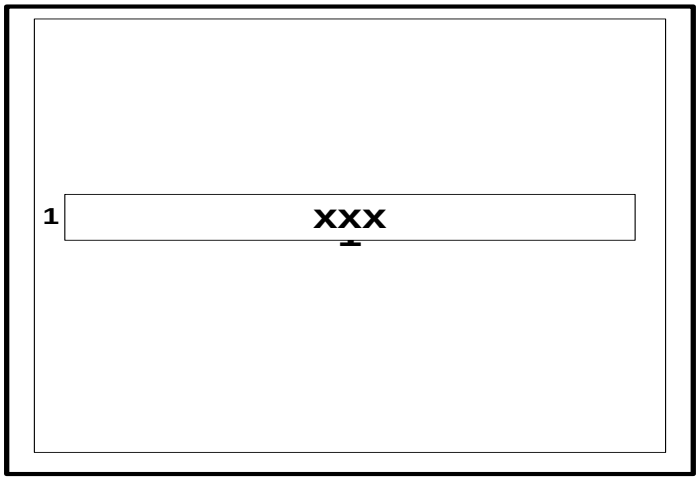
Keterangan Gambar :

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.

3.9 Perancangan Antarmuka

1. Rancangan Halaman Judul

Halaman judul merupakan halaman yang pertama muncul pada saat program dijalankan



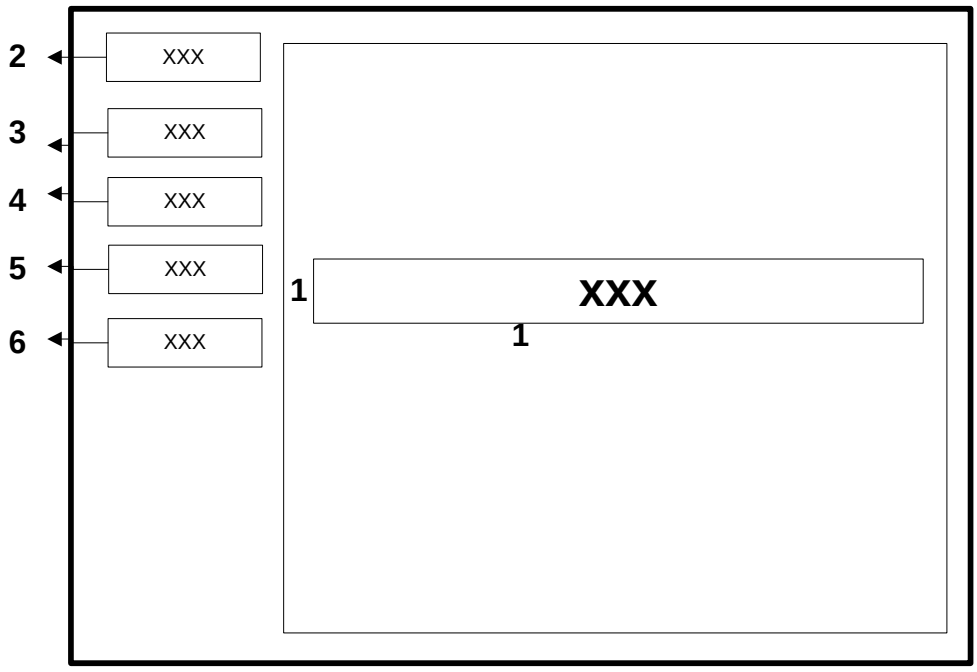
Gambar 3.8 Rancangan Halaman Judul

Pada rancangan di atas akan menampilkan judul yang kemudian akan pindah ke form menu utama dengan menggunakan timer.

Keterangan:

1. Berfungsi untuk menampilkan judul program.
2. Rancangan Halaman Menu Utama

Form ini berisi tombol-tombol seperti menu Materi, Enkripsi, Deskipsi, tentang, dan Keluar.



Gambar 3.9 Rancangan Halaman Menu Utama

Pada tampilan di atas terdapat 5 tombol yaitu Materi, Enkripsi, Deskripsi, Tabel Affine, Tentang dan keluar.

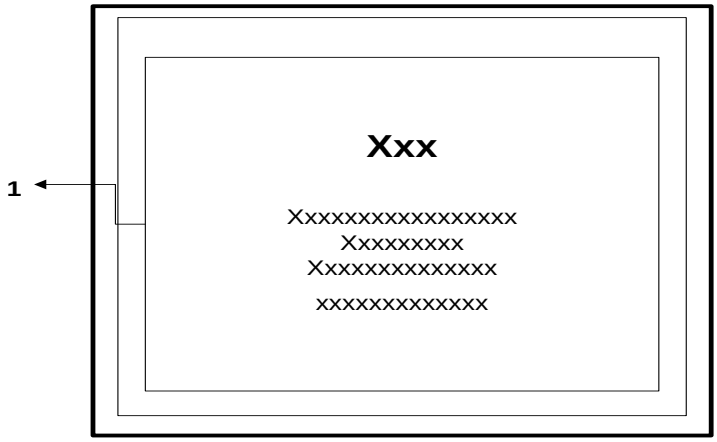
Keterangan:

1. Tombol Berfungsi untuk menampilkan judul program.
2. Tombol Materi berfungsi untuk menghubungkan pengguna ke form materi.
3. Tombol Enkripsi berfungsi untuk menghubungkan pengguna ke form Enkripsi.
4. Tombol Deskripsi berfungsi untuk menampilkan form Deskripsi.
5. Tombol Tentang berfungsi untuk menghubungkan pengguna ke form tentang.
6. Tombol Keluar berfungsi untuk keluar dari program.

3. Rancangan Halaman Materi

   Form ini digunakan untuk menjelaskan cara kerja penyandian, dimulai dari plaintext kemudian kunci yang dikonversikan dalam bentuk angka. Setelah itu dilakukan proses

penjumlahan dan jika hasil penjumlahan maka akan dikurangi 6 lalu hasilnya akan dikembalikan lagi ke dalam bentuk huruf.



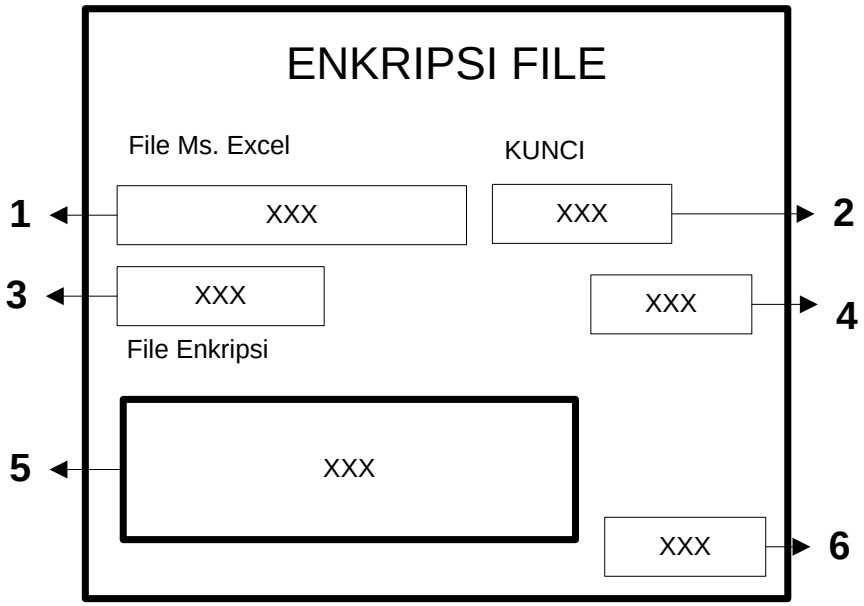
Gambar 3.10 Rancangan Halaman Materi

Keterangan:

1. Tombol Berfungsi untuk menampilkan Materi tentang Kriptografi LSB

2. Rancangan Halaman Enkripsi

Berisi penjelasan mengenai Enkripsi. Pengguna memasukkan tulisan asli atau *plaintext* ke dalam tombol masukan *plaintext* kemudian dimasukkan juga kunci. Setelah itu, ditekan tombol Proses Enkripsi yang kemudian akan menampilkan ciphertext atau tulisan yang telah disandikan.

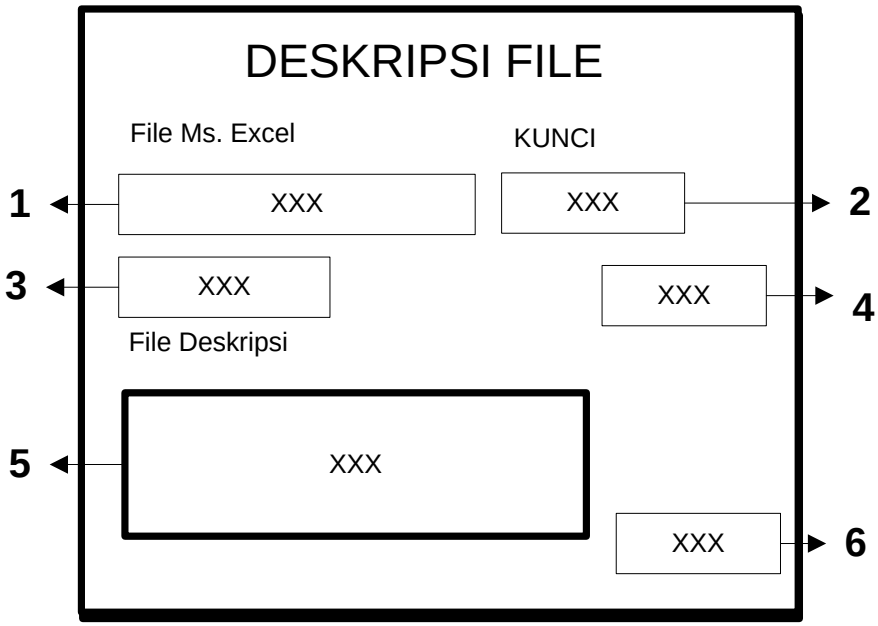


Gambar 3.11 Rancangan Halaman Enkripsi

Keterangan:

1. Berfungsi untuk menampilkan nama Ms. Excel yang sudah di upload.
2. Berfungsi untuk menginputkan kunci untuk mengenkripsi Ms. Excel.
3. Tombol yang berfungsi untuk mencari Ms. Excel yang ingin di enkripsi.
4. Tombol yang berfungsi untuk melakukan proses enkripsi pada Ms. Excel menggunakan LSB.
5. Berfungsi untuk menampilkan hasil dari proses enkripsi dari sebuah Ms. Excel.
6. Deskripsi Rancangan Halaman

Berisi penjelasan mengenai Enkripsi. Pengguna memasukkan tulisan asli atau *plaintext* ke dalam tombol masukan *plaintext* kemudian dimasukkan juga kunci. Setelah itu, ditekan tombol Proses Enkripsi yang kemudian akan menampilkan ciphertext atau tulisan yang telah disandikan.



Gambar 3.12 Rancangan Halaman Deskripsi

Keterangan:

1. Berfungsi untuk menampilkan nama Ms. Excel yang sudah di upload.
2. Berfungsi untuk menginputkan kunci untuk mengdeskripsi Ms. Excel.
3. Tombol yang berfungsi untuk mencari Ms. Excel yang ingin di deskripsi.
4. Tombol yang berfungsi untuk melakukan proses deskripsi pada Ms. Excel menggunakan LSB.
5. Berfungsi untuk menampilkan hasil dari proses deskripsi dari sebuah Ms. Excel.

Pada gambar di atas terdapat kotak input Deskripsi berfungsi untuk memasukkan tulisan yang telah disandikan. Kemudian terdapat tombol Proses Deskripsi untuk mengembalikan ke tulisan asli jika kunci yang dimasukkan sama dengan kunci pada saat penggunaan plaintext.

BAB IV

HASIL PEMBAHASAN

4.1 Implementasi Algoritma

Algoritma adalah urutan langkah untuk menyelesaikan masalah secara sistematis dan logis. Algoritma menawarkan suatu metode dalam menyelesaikan sebuah permasalahan. Algoritma diartikan sebagai urutan langkah dalam menyelesaikan masalah secara sistematis dan logis. Pendekatan secara sistematis dan logis tersebut, menjadikan proses penyelesaian masalah terjaga kebenarannya karena algoritma haruslah benar agar dapat menghasilkan solusi yang benar.

4.1.1 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 LSB:

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 dan Pc) G

vB = C. Gp (Pb dan Pc) B

T1 = Mid i, 1

T2 = Mid i + 1, 1

T3 = Mid i + 2, 1

vM = toDecimal(toLSB(ToBiner(vM), T1))

vH = toDecimal(toLSB(ToBiner(vH), T2))

vB = toDecimal(toLSB(ToBiner(vB), T3))

xpix = xpix + 1

If xpix > xpx Then Exit For

i = i + 3

Next

CT ← LSB Image (gambar yang telah berisi pesan)

4.1.2 Proses Mengambil Data Teks (*Extraction*)

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

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

Output : EP

Proses :

For Pc = 0 To SI.Height - 1

For Pb = 0 To SI.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← File 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.

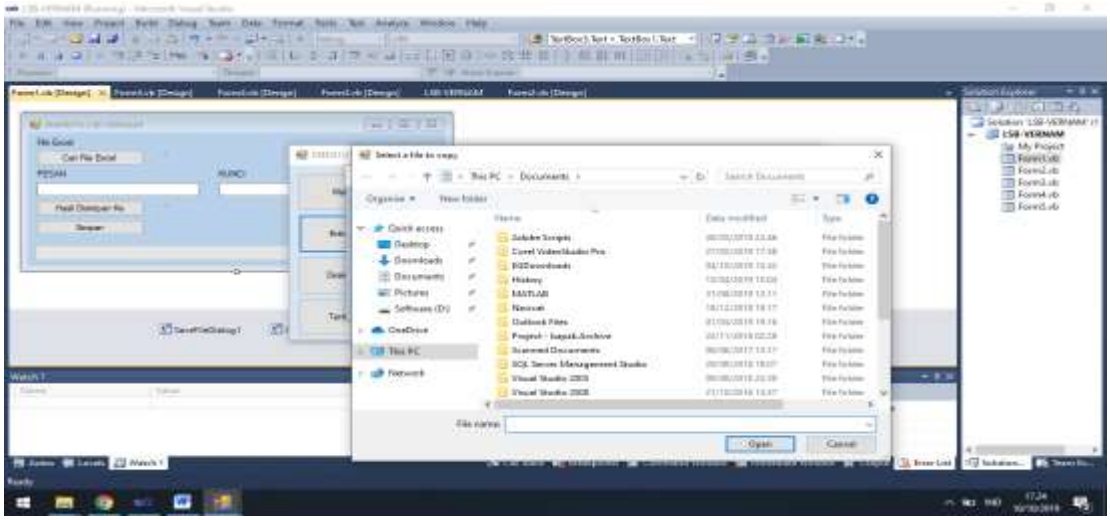
4.2.1 Tampilan Halaman Steganografi

Halaman Steganografi merupakan halaman yang muncul pertama sekali pada saat sistem dijalankan. Tampilan halaman Steganografi dapat dilihat pada Gambar 15.

Gambar 4.1. Tampilan Halaman Menu Utama

4.2.2 Tampilan Cari File

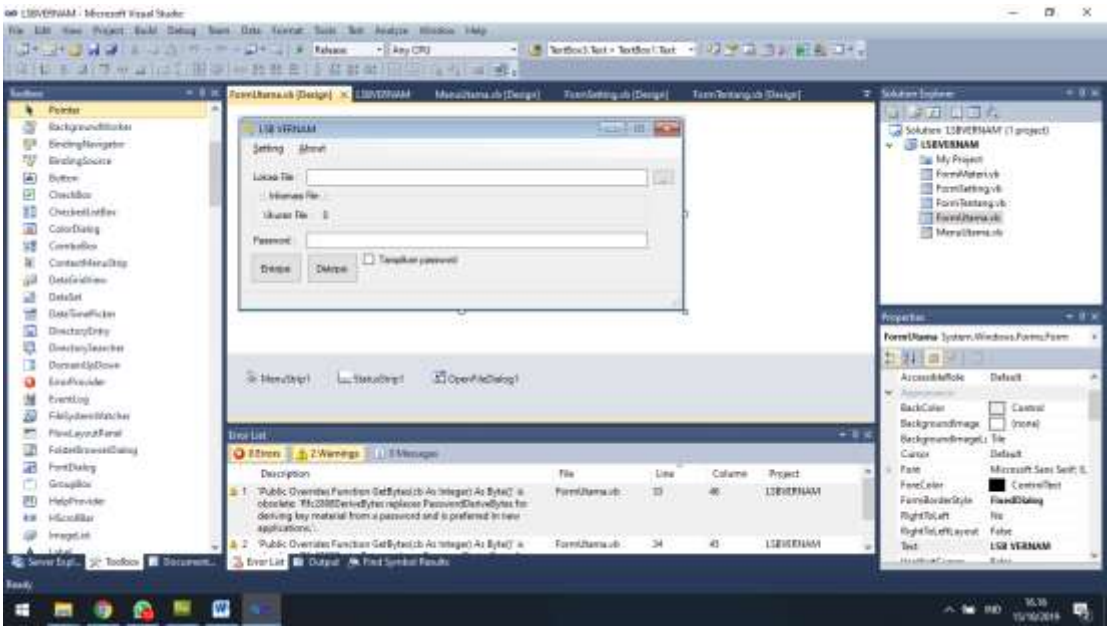
Halaman Cari File merupakan halaman yang muncul pada saat proses untuk penyembunyian pesan. Tampilan halaman Cari File dapat dilihat pada Gambar 4.2.



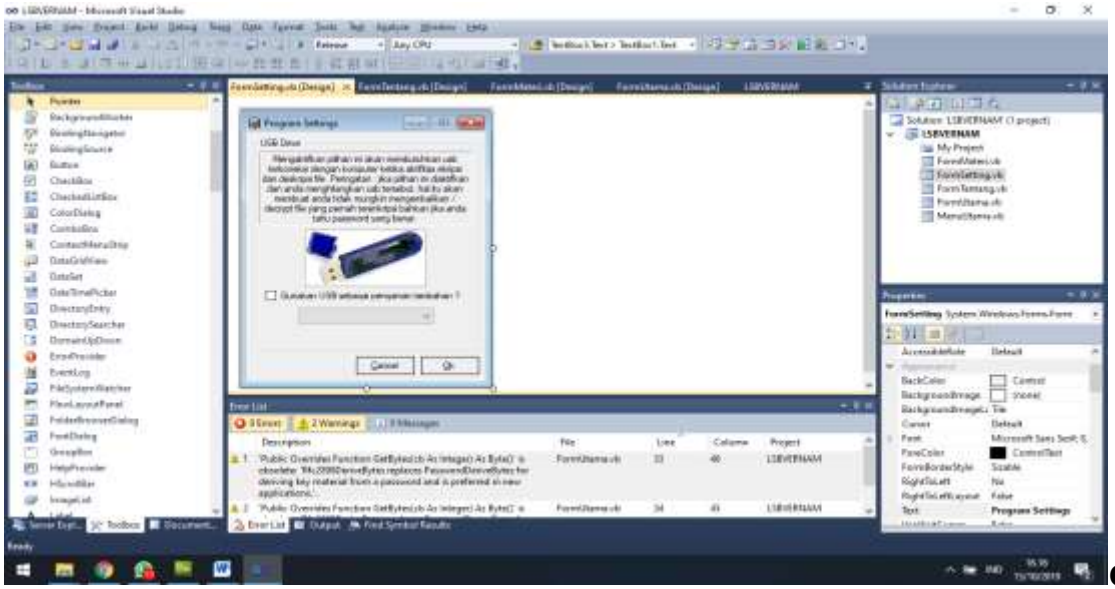
Gambar 4.2. Tampilan Cari File

4.2.3 Tampilan Penyembunyian File

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



Gambar 4.3. Tampilan Penyembunyian Pesan Text



Gambar 4.4. Tampilan Program Setting

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	*Black box*
Mengelola proses penyembunyian pesan text	Input Gambar	*Black box*
	Input Pesan	*Black box*
	Input Password	*Black box*

2) Rencana Pengujian Pengujian Pengguna

Tabel 4.2. Rencana Pengujian Pengguna (*User*)

Menu yang diuji	**Detai pengujian**	**Jenis uji**
Input Password	Menginputkan Key Pada Pesan	*Black box*
Input Gambar	Mencari Gambar Untuk Media Pesan	*Black box*

Input Pesan	Menampilkan Pesan yang ada pada Gambar.	*Black box*

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 (*user*)
Kondisi awal	Berada dihalaman utama
Kondisi akhir	File Gambar Muncul Pada *Picture Box*
Skenario	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 (*user*)
Kondisi awal	Berada pada Menu Utama
Kondisi akhir	Menghasilkan Pesan pada gambar yang sudah tersisipkan text.
Skenario	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 Passoword

Nama fungsi	Text Box (Password)
Tujuan	Untuk menguji apakah proses tersebut sesuai dengan yang diinginkan
Aktor	Pengguna (*user*)
Kondisi awal	Berada pada Menu Utama
Kondisi akhir	Menghasilkan Password pada gambar yang sudah tersisipkan text untuk keamanan pesan.
Skenario	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 (*user*)
Kondisi awal	Berada pada Menu Utama
Kondisi akhir	Menghasilkan pesan text yang dihasilkan dari gambar stego.
Skenario	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

4.4 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
Menampilkan l 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

5.1 Kesimpulan

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

1. Algoritma steganografi *Least Significant Bit* dilakukan dengan menggantikan *bit-bit* pesan rahasia pada *bit* terakhir tiap komponen warna piksel citra. Satu komponen warna citra hanya disisipkan satu *bit* pesan (bernilai 0 atau 1) sehingga ukuran citra tidak berubah.
2. Kecepatan waktu proses bergantung pada besarnya file, panjang kunci dan kecepatan prosessor komputer yang digunakan.

5.2 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

Ariyus, Dony. 2006. *Computer Security*. Yogyakarta: Penerbit Andi.

Arjana, Putu H. dkk. 2012. *Implementasi Enkripsi Data Dengan Algoritma LSB.* Yogyakarta: Seminar Nasional Teknologi Informasi dan Komunikasi 2012 (SENTIKA 2012).

Bishop, Matt. 2005. *Introduction to Computer Security*. Boston: Addison-Wesley.

Christensen, Chris. 2006. *Steganografi and LSB.* http:/www.nku.edu/~christensen/section%2014%20steganografi.pdf. Diakses pada 5 November 2016.

Enterprise, Jubilee. 2017. *Otodidak Visual basic.* Yogakarta : Elex Media Komputindo.

Fachri, Barany. Perancangan Sistem Informasi Iklan Produk Halal Mui Berbasis Mobile Web Menggunakan Multimedia Interaktif. Jurasik (Jurnal Riset Sistem Informasi dan Teknik Informatika), 2018, 3: 98-102.

Fuad, R. N., & Winata, H. N. (2017). Aplikasi keamanan file audio wav (waveform) dengan terapan algoritma RSA. InfoTekJar: Jurnal Nasional Informatika dan Teknologi Jaringan, 1(2), 113-119.

Ginting, G., Fadlina, M., Siahaan, A. P. U., & Rahim, R. (2017). Technical approach of Topsis in decision making. Int. J. Recent Trends Eng. Res, 3(8), 58-64.

Hafni, Layla, and Rismawati Rismawati. "Analisis faktor-faktor internal yang mempengaruhi nilai perusahaan pada perusahaan manufaktur yang terdaftar di BEI 2011-2015." Bilancia: Jurnal Ilmiah Akuntansi 1.3 (2017): 371-382.

Hamdi, Muhammad Nurul, Evi Nurjanah, and Latifah Safitri Handayani. "Community development based onibnu khaldun thought, sebuah interpretasi program pemberdayaan umkm di bank zakat el-zawa." EL MUHASABA: Jurnal Akuntansi (e-journal) 5.2 (2014): 158-180.

Hariyanto, E., & Rahim, R. (2016). Arnold's cat map algorithm in digital image encryption. International Journal of Science and Research (IJSR), 5(10), 1363-1365.

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., Haryati, S., & Yusman, Y. (2018). Aplikasi Kamus Bahasa Jawa Indonesia dengan Algoritma Raita Berbasis Android. Jurnal Teknologi Informasi dan Pendidikan, 11(1), 1-6.

Kromodimoeljo, S. (2009). Teori dan Aplikasi Kriptografi. SPK IT Consulting.

Kurnia, D., Dafitri, H., & Siahaan, A. P. U. (2017). RSA 32-bit Implementation Technique. Int. J. Recent Trends Eng. Res, 3(7), 279-284.

Kurniawan, H. (2018). Pengenalan Struktur Baru untuk Web Mining dan Personalisasi Halaman Web. Jurnal Teknik dan Informatika, 5(2), 13-19.

Leong, Marlon. 2006. *Dari Programmer Untuk Proggrammer Visual Basic*. Yogyakarta: Penerbit Andi.

Martin, Keith. 2012. *Everyday Cryptography*. Oxford: Oxford University Press.

Mulyana, Teady. 2012. *Steganografi Citra Digital Menggunakan Spreadsheet*. Vol: 8 No 2 Agustus 2012.

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.

Pabokory, Fresly Nandar dkk. 2015. *Implementasi LSB Pengamanan Data Pada Pesan Teks, Isi File Gambar Menggunakan Algoritma Advanced Encryption Standard.* Vol: 10 No 1 Februari 2015.

Prabawati, A. (2010). *Tutorial 5 hari belajar Pemograman Visual Basic 2010.* Yogyakarta: Andi.

Rhee, Man Young. 1994. *Library of Congress Cataloging-in-Publication Data*. Singapore: McGraw-Hill Book Co.

Stinson, D. (1995). Cryptography Theory and Practice. Florida: CRC Press.

Sutanto, Edhy. 2004. *Algoritma:Teknik Penyelesaian Permasalahan Untuk Komputasi*. Yogyakarta : Graha Ilmu.

Wahana Komputer. 2003. *Memahami Model Enkripsi dan Security Data*. Yogyakarta: Penerbit Andi.