



**IMPLEMENTASI PENYEMBUNYIAN INFORMASI PADA FILE
AUDIO DENGAN METODE LEAST SIGNIFICANT BIT**

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

SKRIPSI

OLEH:

NAMA : INDRIANI
NPM : 1514370403
PROGRAM STUDI : SISTEM KOMPUTER

FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS PEMBANGUNAN PANCA BUDI
MEDAN
2019

ABSTRAK

INDRIANI

Implementasi Penyembunyian Informasi Pada File Audio Dengan Metode Least Significant Bit 2019

Pengiriman informasi adalah hal yang sangat sering dilakukan. Pengiriman ini dapat dikirim melalui jaringan komputer. Pada zaman sekarang, pengiriman informasi tidak perlu ditulis pada selembar kertas. Pengiriman dapat dikirim dengan bantuan sosial media. Pesan-pesan dapat dituliskan dalam bentuk data digital sehingga pesan tersebut dapat dikirim kapan saja dan secara berulang tanpa harus mengeluarkan tenaga yang besar. Tetapi, dengan kecanggihan teknologi zaman sekarang, pengiriman informasi semakin rentan. Informasi tersebut dapat disalahgunakan. Pesan tersebut harus dirahasiakan. Teknik steganografi dapat membantu pengiriman pesan dengan cara tersembunyi. Pesan akan disembunyikan dalam suatu media suara. Algoritma yang digunakan adalah LSB. Algoritma ini bekerja dengan cara mengganti posisi bit terakhir pada sinyal suara dengan bit-bit pesan yang akan disisip. Setiap byte pesan akan menempati delapan buah sinyal suara. Dengan menerapkan algoritma LSB, pengiriman pesan akan berjalan dengan lancar dan rahasia.

Kata Kunci: *algoritma, keamanan, embed, ekstrak, steganografi, LSB*

DAFTAR ISI

| | |
|---|-----------|
| KATA PENGANTAR | i |
| DAFTAR ISI | ii |
| DAFTAR GAMBAR | iv |
| DAFTAR TABEL | v |
| BAB I PENDAHULUAN | 1 |
| 1.1 Latar Belakang | 1 |
| 1.2 Rumusan Masalah | 2 |
| 1.3 Batasan Masalah..... | 2 |
| 1.4 Tujuan Penelitian | 3 |
| 1.5 Manfaat Penelitian | 3 |
| BAB II LANDASAN TEORI | 4 |
| 2.1 Pengertian Aplikasi | 4 |
| 2.2 Data | 4 |
| 2.2.1 Bagaimana Data Disimpan | 6 |
| 2.2.2 Jenis data | 6 |
| 2.2.3 Pengelolaan dan Penggunaan Data..... | 7 |
| 2.3 Keamanan Data | 8 |
| 2.3.1 Pentingnya Keamanan Data | 9 |
| 2.3.2 Solusi Keamanan Data | 11 |
| 2.3.3 Kerahasiaan | 12 |
| 2.3.4 Integritas | 13 |
| 2.3.5 Ketersediaan | 14 |
| 2.3.6 Kontrol Akses | 14 |
| 2.4 Algoritma | 15 |
| 2.4.1 Desain Konseptual..... | 16 |
| 2.4.2 Tugas Algoritma..... | 17 |
| 2.4.3 Rekayasa Algoritma | 18 |
| 2.5 Steganografi | 18 |
| 2.5.1 Sejarah Steganografi..... | 22 |
| 2.5.2 Stego File..... | 23 |
| 2.6 Audio..... | 23 |
| 2.7 Least Significant Bit..... | 25 |
| 2.8 Unified Modelling Language (UML)..... | 29 |
| 2.8.1 Use Case Diagram | 29 |
| 2.8.2 Activity Diagram | 33 |
| 2.9 Flowchart..... | 34 |
| 2.10 Visual Basic | 37 |
| 2.10.1 Lingkungan kerja Visual Basic.Net..... | 38 |
| 2.10.2 Komponen Visual Basic.Net | 39 |
| BAB III METODE PENELITIAN | 43 |

| | | |
|---|---|-----------|
| 3.1 | Kerangka Penelitian | 43 |
| a. | Perancangan Penelitian | 45 |
| | i. Use Case Diagram | 46 |
| | ii. Activity Diagram | 46 |
| | iii. Flowchart Penyisipan | 48 |
| | iv. Flowchart Ekstraksi | 49 |
| b. | Desain Interface | 50 |
| | i. Menu Utama | 50 |
| | ii. Menu LSB | 51 |
| | iii. Menu Abstrak | 52 |
| | iv. Menu About | 53 |
| BAB IV HASIL DAN PEMBAHASAN..... | | 54 |
| 4.1 | Spesifikasi Sistem | 54 |
| | 4.1.1 Spesifikasi Perangkat Keras | 55 |
| | 4.1.2 Spesifikasi Perangkat Lunak | 55 |
| 4.2 | Implementasi Antarmuka | 56 |
| | 4.2.1 Halaman Menu Utama | 56 |
| | 4.2.2 Halaman Abstrak | 57 |
| | 4.2.3 Halaman About | 57 |
| | 4.2.4 Halaman LSB | 58 |
| | 4.2.5 Hasil Penyisipan | 59 |
| | 4.2.6 Hasil Ekstraksi | 60 |
| BAB V PENUTUP..... | | 62 |
| 5.1 | Kesimpulan | 62 |
| 5.2 | Saran | 62 |
| DAFTAR PUSTAKA | | 1 |

DAFTAR GAMBAR

| | |
|--|----|
| Gambar 2.1 Skema Steganografi..... | 20 |
| Gambar 2.2 Contoh susunan bit pada <i>LSB</i> dan <i>MSB</i> | 26 |
| Gambar 2.3 Use-case Diagram ATM..... | 31 |
| Gambar 2.4 Tampilan Microsoft Visual Studio 2010 | 38 |
| Gambar 2.5 Tampilan Menu Bar | 39 |
| Gambar 2.6 Tampilan Toolbar | 39 |
| Gambar 2.7 Tampilan Toolbox | 40 |
| Gambar 2.8 Tampilan Properties | 41 |
| Gambar 2.9 Tampilan Form | 42 |
| Gambar 2.10 Tampilan Code Editor | 42 |
| Gambar 3.1 Kerangka Penelitian | 44 |
| Gambar 3.2 Use Case Diagram | 46 |
| Gambar 3.3 Activity Diagram penyisipan dan ekstraksi pesan | 47 |
| Gambar 3.4 Flowchart penyisipan pesan | 48 |
| Gambar 3.5 Flowchart ekstraksi pesan | 49 |
| Gambar 3.6 Tampilan Menu Utama..... | 50 |
| Gambar 3.7 Tampilan Menu LSB | 51 |
| Gambar 3.8 Tampilan Abstrak | 52 |
| Gambar 3.9 Tampilan Menu About | 53 |
| Gambar 4.1 Halaman Menu Utama | 56 |
| Gambar 4.2 Halaman Abstrak | 57 |
| Gambar 4.3 Halaman About | 58 |
| Gambar 4.4 Halaman menu LSB | 59 |
| Gambar 4.5 Hasil penyisipan pesan ke dalam file | 60 |
| Gambar 4.6 Hasil ekstraksi pesan dari file..... | 61 |

DAFTAR TABEL

| | |
|---|----|
| Tabel 2.1 Simbol Use Case Diagram | 32 |
| Tabel 2.2 Simbol Activity Diagram | 34 |
| Tabel 2.3 Simbol Flowchart | 36 |
| Tabel 4.1 Spesifikasi perangkat keras | 55 |
| Tabel 4.2 Spesifikasi perangkat lunak | 55 |

KATA PENGANTAR

Puji syukur kehadirat Allah SWT, karena dengan berkat dan rahmatNya penulis masih diberikan kesempatan untuk menyelesaikan skripsi ini sebagaimana mestinya. Skripsi ini berjudul **"Implementasi Penyembunyian Informasi Pada File Audio Dengan Metode Least Significant Bit"**. Dalam kesempatan ini, penulis mengucapkan rasa terima kasih yang tak terhingga kepada pihak-pihak yang telah membantu dalam penyelesaian skripsi ini. Penulis ingin mengucapkan terima kasih kepada :

1. Orang tua saya yang selalu memberikan semangat, dukungan dan motivasi dalam penyusunan skripsi ini.
2. Bapak Dr. H. Muhammad Isa Indrawan, S.E, M.M., selaku Rektor Universitas Pembangunan Panca Budi Medan.
3. Rektor I Bapak Ir. Bhakti Alamsyah, M.T, Ph.D.
4. Bapak Hamdani, ST., M.T., selaku Dekan Fakultas Sains dan Teknologi Universitas Pembangunan Panca Budi Medan.
5. Bapak Eko Hariyanto, S.Kom., M.Kom, selaku Ketua Program Studi Sistem Komputer Universitas Pembangunan Panca Budi Medan.
6. Bapak Andysah Putera Utama Siahaan, S.Kom., M.Kom., Ph.D., selaku Dosen Pembimbing I yang telah memberikan arahan dan membimbing dalam penyelesaian skripsi ini.
7. Ibu Virdyra Tasril, S.Kom., M.Kom., selaku Dosen Pembimbing II yang telah memberikan ilmu pengetahuan, serta bimbingan dalam penyelesaian skripsi ini.
8. Dosen-dosen pada Program Studi Sistem Komputer Fakultas Sains dan Teknologi Universitas Pembangunan Panca Budi Medan.
9. Staff dan karyawan pada Universitas Pembangunan Panca Budi Medan.
10. Seluruh teman-teman penulis dari program studi Sistem Komputer, Fakultas Ilmu Komputer Universitas Pembangunan Panca Budi, Medan

Penulis juga menyadari bahwa penyusunan skripsi ini belum mendapatkan kesempurnaan dalam segi penulisan ataupun isi. Hal ini disebabkan pengetahuan penulis yang sangat terbatas. Penulis sangat mengharapkan adanya kritik dan saran dari pembaca untuk dapat memperbaiki isi skripsi.

Medan, 23 Desember 2019
Penulis

Indriani
1514370403

BAB I

PENDAHULUAN

1.1 Latar Belakang

Informasi tidak luput dari pengiriman. Informasi perlu dipertukarkan dengan orang lain karena orang tersebut tidak berada dekat dengan pengirim pesan tersebut. Jika orang tersebut ada di dekat pengirim, maka pesan tidak perlu dikirimkan karena sudah langsung dapat didengar oleh penerima pesan. Pengiriman pesan membutuhkan media atau transmisi. Dalam melakukan pengiriman pesan, ada hal-hal yang dapat terjadi pada saat pesan tersebut melintas diudara. Pesan tersebut dikirimkan secara nirkabel dan bebas mengudara. Tetapi hal ini dapat menjadikan pesan tersebut rentan terhadap pencurian dan penyalahgunaan data. Pesan yang sedang melintas diudara tidak memiliki kekuatan apa-apa. Sinyal-sinyal frekuensi pesan tersebut dapat dengan mudah dicuri dan dibentuk kembali sehingga menghasilkan rangkaian bit-bit yang dapat dikembalikan lagi atau disusun kembali menjadi pesan seperti semula.

Dalam melakukan pengamanan informasi, teknik steganografi dapat digunakan untuk mengelabui pencuri pesan terhadap apa yang akan diambilnya. Pesan yang akan dikirim akan dikemas dan ditanamkan kedalam media lain. Media yang dapat digunakan adalah gambar, suara, video atau file-file lainnya. Dalam penelitian ini, penulis mencoba menanamkan informasi ke dalam file suara berbentuk lagu. Algoritma yang digunakan adalah Least Significant Bit (LSB). Algoritma ini bekerja dengan cara mengganti bit-bit pertama pada sinyal suara

dengan bit-bit pesan yang akan dikirimkan. Setiap karakter pesan akan menempati sejumlah delapan karakter sinyal suara.

Oleh karena itu, penelitian tentang steganografi akan selalu berkembang sehingga menghasilkan suatu penelitian baru. Berdasarkan latar belakang tentang steganografi yang sudah dipaparkan, maka penulis ingin mengambil topik keamanan informasi dengan judul **“Implementasi Penyembunyian Informasi Pada File Audio Dengan Metode Least Significant Bit”**.

1.2 Rumusan Masalah

Adapun rumusan masalah yang digunakan dalam penulisan tugas akhir ini adalah sebagai berikut:

1. Bagaimana melakukan proses penyisipan pada file audio dengan menggunakan algoritma LSB?
2. Bagaimana menentukan pergantian bit pertama pada sinyal suara?
3. Bagaimana melakukan proses ekstraksi pada file audio dengan menggunakan algoritma LSB?

1.3 Batasan Masalah

Adapun batasan masalah yang digunakan dalam penulisan tugas akhir ini adalah sebagai berikut:

1. Media yang digunakan adalah file audio .mp3 dan .wav.
2. Panjang pesan digunakan adalah sebanyak 1000 karakter.
3. Pesan yang akan disisipkan diketikkan langsung pada textbox pesan.
4. Program aplikasi menggunakan Microsoft Visual Basic.Net 2010.

1.4 Tujuan Penelitian

Adapun tujuan penelitian yang digunakan dalam penulisan tugas akhir ini adalah sebagai berikut:

1. Untuk melakukan proses penyisipan pada file audio dengan menggunakan algoritma LSB.
2. Untuk menentukan pergantian bit pertama pada sinyal suara.
3. Untuk melakukan proses ekstraksi pada file audio dengan menggunakan algoritma LSB.

1.5 Manfaat Penelitian

Adapun manfaat penelitian yang digunakan dalam penulisan tugas akhir ini adalah sebagai berikut:

1. Mengamankan pesan pada saat pengiriman.
2. Pesan tersembunyi tidak mudah ditemukan sehingga menghindari penyalahgunaan data.
3. Memberikan kenyamanan pengirim dan penerima pesan.

BAB II

LANDASAN TEORI

2.1 Pengertian Aplikasi

Secara istilah pengertian aplikasi adalah suatu program yang siap untuk digunakan yang dibuat untuk melaksanakan suatu fungsi bagi pengguna jasa aplikasi serta penggunaan aplikasi lain yang dapat digunakan oleh suatu sasaran yang akan dituju. Menurut kamus komputer eksekutif, aplikasi mempunyai arti yaitu pemecahan masalah yang menggunakan salah satu tehnik pemrosesan data aplikasi yang biasanya berpacu pada sebuah komputansi yang diinginkan atau diharapkan maupun pemrosesan data yang di harapkan.

Menurut Kamus Kamus Besar Bahasa Indonesia, aplikasi adalah penerapan dari rancang sistem untuk mengolah data yang menggunakan aturan atau ketentuan bahasa pemrograman tertentu. Aplikasi adalah suatu program komputer yang dibuat untuk mengerjakan dan melaksanakan tugas khusus dari pengguna. Aplikasi merupakan rangkaian kegiatan atau perintah untuk dieksekusi oleh komputer.

2.2 Data

Data, dalam konteks komputasi, mengacu pada bagian informasi digital yang berbeda. Data biasanya diformat dengan cara tertentu dan dapat ada dalam berbagai bentuk, seperti angka, teks, dll. Ketika digunakan dalam konteks media transmisi, data merujuk ke informasi dalam format digital biner. Data adalah istilah luas dalam teknologi komputer, tetapi sering digunakan untuk mengidentifikasi dan

memisahkan informasi dari bit belaka. Dalam telekomunikasi, data sering merujuk pada informasi digital, bukan analog. Tidak seperti transmisi analog, yang memerlukan koneksi garis keras selama durasi transmisi, data digital dikirim dalam paket (Sun, Zhang, Xiong, & Zhu, 2014).

Dalam komputasi, data adalah informasi yang telah diterjemahkan ke dalam bentuk yang efisien untuk pergerakan atau pemrosesan. Relatif terhadap komputer dan media transmisi saat ini, data adalah informasi yang diubah menjadi bentuk digital biner. Data dapat diterima untuk digunakan sebagai subjek tunggal atau subjek jamak. Data mentah adalah istilah yang digunakan untuk menggambarkan data dalam format digital paling dasar.

Konsep data dalam konteks komputasi berakar pada karya Claude Shannon, seorang ahli matematika Amerika yang dikenal sebagai bapak teori informasi. Dia mengantarkan konsep digital biner berdasarkan penerapan logika Boolean dua nilai ke sirkuit elektronik. Format digit biner mendasari CPU, memori semikonduktor dan disk drive, serta banyak perangkat periferifal yang umum dalam komputasi saat ini. Input komputer awal untuk kontrol dan data berupa kartu punch, diikuti oleh pita magnetik dan hard disk.

Pada awalnya, pentingnya data dalam komputasi bisnis menjadi jelas dengan popularitas istilah "pemrosesan data" dan "pemrosesan data elektronik," yang, untuk beberapa waktu, datang untuk mencakup keseluruhan dari apa yang sekarang dikenal sebagai teknologi informasi. Selama sejarah komputasi perusahaan, spesialisasi terjadi, dan profesi data yang berbeda muncul seiring dengan pertumbuhan pemrosesan data perusahaan.

2.2.1 Bagaimana Data Disimpan

Komputer mewakili data, termasuk video, gambar, suara dan teks, sebagai nilai biner menggunakan pola hanya dua angka: 1 dan 0. Sedikit adalah unit data terkecil dan hanya mewakili nilai tunggal. Satu byte terdiri dari delapan digit biner. Penyimpanan dan memori diukur dalam megabita dan gigabita.

Unit-unit pengukuran data terus bertambah seiring dengan meningkatnya jumlah data yang dikumpulkan dan disimpan. Istilah "brontobyte" yang relatif baru, misalnya, adalah penyimpanan data yang setara dengan 10 hingga 27 byte. Data dapat disimpan dalam format file, seperti pada sistem mainframe menggunakan ISAM dan VSAM. Format file lain untuk penyimpanan, konversi, dan pemrosesan data termasuk nilai yang dipisah koma. Format ini terus menemukan kegunaan di berbagai jenis mesin, bahkan ketika pendekatan yang lebih berorientasi data terstruktur memperoleh pijakan dalam komputasi perusahaan. Spesialisasi yang lebih besar dikembangkan sebagai basis data, sistem manajemen basis data, dan kemudian teknologi basis data relasional muncul untuk mengatur informasi (Zhang et al., 2009).

2.2.2 Jenis data

Pertumbuhan web dan telepon pintar selama dekade terakhir menyebabkan peningkatan dalam penciptaan data digital. Data sekarang termasuk informasi teks, audio dan video, serta catatan aktivitas log dan web. Banyak dari itu adalah data yang tidak terstruktur.

Istilah big data telah digunakan untuk menggambarkan data dalam kisaran petabyte atau lebih besar. Tulisan singkat menggambarkan data besar dengan 3V - volume, variasi, dan kecepatan. Ketika e-commerce berbasis web telah menyebar, model bisnis berbasis data besar telah berevolusi yang memperlakukan data sebagai aset. Tren semacam itu juga telah menimbulkan keasyikan yang lebih besar dengan penggunaan sosial data dan privasi data.

Data memiliki makna di luar penggunaannya dalam aplikasi komputasi yang berorientasi pada pemrosesan data. Misalnya, dalam interkoneksi komponen elektronik dan komunikasi jaringan, istilah data sering dibedakan dari "informasi kontrol," "bit kontrol," dan istilah serupa untuk mengidentifikasi konten utama dari unit transmisi. Selain itu, dalam sains, istilah data digunakan untuk menggambarkan kumpulan fakta. Itu juga terjadi di bidang-bidang seperti keuangan, pemasaran, demografi dan kesehatan.

2.2.3 Pengelolaan dan Penggunaan Data

Dengan semakin banyaknya data dalam organisasi, penekanan tambahan telah ditempatkan pada memastikan kualitas data dengan mengurangi duplikasi dan menjamin yang paling akurat, catatan saat ini digunakan. Banyak langkah yang terlibat dengan manajemen data modern termasuk pembersihan data, serta mengekstrak, mengubah dan memuat (ETL) proses untuk mengintegrasikan data. Data untuk diproses telah dilengkapi dengan metadata, kadang-kadang disebut sebagai "data tentang data," yang membantu administrator dan pengguna memahami database dan data lainnya.

Analisis yang menggabungkan data terstruktur dan tidak terstruktur menjadi bermanfaat, karena organisasi berupaya memanfaatkan informasi tersebut. Sistem untuk analitik semacam itu semakin berupaya untuk kinerja waktu-nyata, sehingga mereka dibangun untuk menangani data yang masuk yang dikonsumsi dengan tingkat konsumsi tinggi, dan untuk memproses aliran data untuk penggunaan langsung dalam operasi.

Seiring waktu, gagasan basis data untuk operasi dan transaksi telah diperluas ke basis data untuk pelaporan dan analitik data prediktif. Contoh utama adalah gudang data, yang dioptimalkan untuk memproses pertanyaan tentang operasi untuk analis bisnis dan pemimpin bisnis. Meningkatnya penekanan pada menemukan pola dan memprediksi hasil bisnis telah mengarah pada pengembangan teknik penambangan data (Barone, Williams, & Micklos, 2017).

2.3 Keamanan Data

Keamanan data adalah seperangkat standar dan teknologi yang melindungi data dari kehancuran, modifikasi, atau pengungkapan yang disengaja atau tidak disengaja. Keamanan data dapat diterapkan dengan menggunakan berbagai teknik dan teknologi, termasuk kontrol administratif, keamanan fisik, kontrol logis, standar organisasi, dan teknik perlindungan lainnya yang membatasi akses ke pengguna atau proses yang tidak sah atau berbahaya (Rao & Selvamani, 2015).

Keamanan data mengacu pada langkah-langkah privasi digital pelindung yang diterapkan untuk mencegah akses tidak sah ke komputer, database, dan situs web. Keamanan data juga melindungi data dari korupsi. Keamanan data adalah

aspek penting dari TI untuk organisasi dari berbagai ukuran dan tipe. Keamanan data juga dikenal sebagai keamanan informasi atau keamanan komputer.

Contoh teknologi keamanan data termasuk backup, masking data dan penghapusan data. Ukuran teknologi keamanan data utama adalah enkripsi, di mana data digital, perangkat lunak / perangkat keras, dan hard drive dienkripsi dan karenanya tidak dapat dibaca oleh pengguna dan peretas yang tidak sah. Salah satu metode yang paling umum dijumpai dalam mempraktikkan keamanan data adalah penggunaan otentikasi. Dengan otentikasi, pengguna harus memberikan kata sandi, kode, data biometrik, atau bentuk data lainnya untuk memverifikasi identitas sebelum akses ke sistem atau data diberikan. Keamanan data juga sangat penting untuk catatan perawatan kesehatan, sehingga pendukung kesehatan dan praktisi medis di AS dan negara-negara lain berupaya menerapkan privasi rekam medis elektronik dengan menciptakan kesadaran tentang hak-hak pasien terkait dengan pelepasan data ke laboratorium, dokter, rumah sakit dan fasilitas medis lainnya.

2.3.1 Pentingnya Keamanan Data

Semua bisnis saat ini menangani data hingga taraf tertentu. Dari raksasa perbankan yang menangani data pribadi dan keuangan dalam volume besar hingga bisnis satu orang yang menyimpan detail kontak pelanggannya di ponsel, data berperan di perusahaan baik besar maupun kecil.

Tujuan utama keamanan data adalah untuk melindungi data yang dikumpulkan, disimpan, diterima, atau ditransmisikan oleh suatu organisasi. Kepatuhan juga merupakan pertimbangan utama. Tidak masalah perangkat,

teknologi, atau proses mana yang digunakan untuk mengelola, menyimpan, atau mengumpulkan data, itu harus dilindungi. Pelanggaran data dapat menyebabkan kasus litigasi dan denda yang sangat besar, belum lagi kerusakan reputasi organisasi. Pentingnya melindungi data dari ancaman keamanan lebih penting saat ini daripada sebelumnya.

Keamanan data mengacu pada proses melindungi data dari akses yang tidak sah dan korupsi data sepanjang siklus hidupnya. Keamanan data termasuk enkripsi data, penyembunyian data, dan praktik manajemen kunci yang melindungi data di semua aplikasi dan platform. Organisasi di seluruh dunia banyak berinvestasi dalam kemampuan pertahanan cyber teknologi informasi untuk melindungi aset penting mereka. Apakah suatu perusahaan perlu melindungi merek, modal intelektual, dan informasi pelanggan atau menyediakan kontrol untuk infrastruktur penting, sarana untuk mendeteksi insiden dan merespons melindungi kepentingan organisasi memiliki tiga elemen umum: orang, proses, dan teknologi.

2.3.2 Solusi Keamanan Data

Data membutuhkan keamanan dalam mengamankan informasi yang ada dalam data tersebut. Dengan keamanan data canggih, tokenization, dan manajemen utama untuk melindungi data di seluruh aplikasi, transaksi, penyimpanan, dan platform big data, Teknik ini menyederhanakan perlindungan data sensitif bahkan dalam kasus penggunaan yang paling kompleks sekalipun. Beberapa model keamanan data antara lain:

1. Keamanan akses cloud - Platform perlindungan yang memungkinkan Anda untuk pindah ke cloud dengan aman sambil melindungi data dalam aplikasi cloud.
2. Enkripsi data - Solusi keamanan data-sentris dan tokenisasi yang melindungi data di lingkungan perusahaan, cloud, seluler, dan data besar.
3. Modul keamanan perangkat keras - Modul keamanan perangkat keras yang menjaga data keuangan dan memenuhi persyaratan keamanan dan kepatuhan industri.
4. Manajemen kunci - Solusi yang melindungi data dan memungkinkan kepatuhan regulasi industri.
5. Enterprise Data Protection - Solusi yang menyediakan pendekatan data-centric end-to-end untuk perlindungan data perusahaan.
6. Keamanan Pembayaran - Solusi menyediakan enkripsi dan tokenisasi point-to-point lengkap untuk transaksi pembayaran ritel, memungkinkan pengurangan lingkup PCI.
7. Big Data, Hadoop, dan perlindungan data IofT - Solusi yang melindungi data sensitif di Danau Data - termasuk Hadoop, Teradata, Micro Focus Vertica, dan platform Big Data lainnya.
8. Keamanan Aplikasi Seluler - Melindungi data sensitif di aplikasi seluler asli sembari menjaga data dari ujung ke ujung.
9. Keamanan Peramban Web - Melindungi data sensitif yang diambil di peramban, dari titik pelanggan memasukkan pemegang kartu atau data

pribadi dan menjaganya agar tetap terlindungi melalui ekosistem ke tujuan tuan rumah tepercaya.

10. eMail Security - Solusi yang menyediakan enkripsi ujung ke ujung untuk email dan olahpesan seluler, menjaga informasi pribadi dan informasi kesehatan pribadi tetap aman dan pribadi.

2.3.3 Kerahasiaan

Kerahasiaan mengacu pada melindungi informasi agar tidak diakses oleh pihak yang tidak berwenang. Dengan kata lain, hanya orang yang diberi wewenang untuk melakukannya yang dapat memperoleh akses ke data sensitif. Bayangkan catatan bank harus dapat diakses, tentu saja, dan karyawan di bank yang membantu dalam menjalankan transaksi harus dapat mengaksesnya, tetapi tidak ada orang lain yang seharusnya. Kegagalan untuk menjaga kerahasiaan berarti bahwa seseorang yang seharusnya tidak memiliki akses telah berhasil mendapatkannya, melalui perilaku yang disengaja atau karena kecelakaan. Kegagalan kerahasiaan seperti itu, umumnya dikenal sebagai pelanggaran, biasanya tidak dapat diperbaiki. Setelah rahasia itu terungkap, tidak ada cara untuk mengetahuinya. Jika catatan bank diposting di situs web publik, semua orang dapat mengetahui nomor rekening bank, saldo, dll., Informasi itu tidak dapat dihapus dari pikiran, kertas, komputer, dan tempat lain mereka. Hampir semua insiden keamanan utama yang dilaporkan di media saat ini melibatkan kerugian besar kerahasiaan. Jadi, secara ringkas, pelanggaran kerahasiaan berarti bahwa seseorang memperoleh akses ke informasi yang seharusnya tidak memiliki akses ke sana.

2.3.4 Integritas

Integritas mengacu pada memastikan keaslian informasi — bahwa informasi tidak diubah, dan bahwa sumber informasi itu asli. Bayangkan jika seseorang memiliki situs web dan Anda menjual produk di situs itu. Sekarang bayangkan penyerang dapat berbelanja di situs web dan dengan jahat mengubah harga produk Anda sehingga mereka dapat membeli apa pun dengan harga berapa pun yang mereka pilih. Itu akan menjadi kegagalan integritas karena informasi dalam hal ini, harga suatu produk telah diubah dan perubahan ini tidak dapat digagalkan. Contoh lain dari kegagalan integritas adalah ketika seseorang mencoba terhubung ke situs web dan penyerang jahat antara Anda dan situs web mengalihkan lalu lintas ke situs web yang berbeda. Dalam hal ini, situs yang dituju tidak asli.

2.3.5 Ketersediaan

Ketersediaan berarti informasi dapat diakses oleh pengguna yang berwenang. Jika penyerang tidak dapat mengkompromikan dua elemen pertama dari keamanan informasi (lihat di atas) mereka dapat mencoba melakukan serangan seperti penolakan layanan yang akan menurunkan server, membuat situs web tidak tersedia untuk pengguna yang sah karena kurangnya ketersediaan.

2.3.6 Kontrol Akses

Kesalahan terbesar yang bisa dilakukan oleh perancang aplikasi adalah mengabaikan kontrol akses sebagai bagian dari fungsionalitas yang diperlukan.

Jarang bahwa setiap pengguna atau sistem yang berinteraksi dengan suatu aplikasi harus memiliki hak yang sama di seluruh aplikasi itu. Beberapa pengguna mungkin memerlukan akses ke data tertentu dan bukan yang lain; beberapa sistem harus atau tidak dapat mengakses aplikasi. Akses ke komponen, fungsi, atau modul tertentu dalam aplikasi juga harus dikontrol. Kontrol akses juga penting untuk kepatuhan audit dan peraturan. Beberapa cara umum mengelola kontrol akses adalah:

1. Baca, tulis, dan jalankan hak istimewa: File
2. Kontrol akses berbasis peran: administrator, pengguna
3. Alamat IP akses berbasis host, nama mesin
4. Objek kode kontrol akses tingkat objek, banyak pembaca / penulis tunggal

2.4 Algoritma

Pengertian algoritma sangat lekat dengan kata logika, yaitu kemampuan seorang manusia untuk berfikir dengan akal tentang suatu permasalahan menghasilkan sebuah kebenaran, dibuktikan dan dapat diterima akal, logika seringkali dihubungkan dengan kecerdasan, seseorang yang mampu berlogika dengan baik sering orang menyebutnya sebagai pribadi yang cerdas (Gurevich, 2012). Logika identik dengan masuk akal dan penalaran. Penalaran adalah salah satu bentuk pemikiran. Pemikiran adalah pengetahuan tak langsung yang didasarkan pada pernyataan langsung pemikiran mungkin benar dan mungkin juga tak benar. Definisi logika sangat sederhana yaitu ilmu yang memberikan prinsip-prinsip yang harus diikuti agar dapat berfikir valid menurut aturan yang berlaku. Pelajaran logika menimbulkan kesadaran untuk menggunakan prinsip-prinsip untuk berfikir

secara sistematis. Logika berasal dari bahasa Yunani yaitu LOGOS yang berarti ilmu. Logika dapat diartikan ilmu yang mengajarkan cara berpikir untuk melakukan kegiatan dengan tujuan tertentu. Algoritma berasal dari nama seorang Ilmuwan Arab yang bernama Abu Jafar Muhammad Ibnu Musa Al Khuwarizmi penulis buku berjudul Al Jabar Wal Muqabala. Kata Al Khuwarizmi dibaca orang barat menjadi Algorism yang kemudian lambat laun menjadi Algorithm diserap dalam bahasa Indonesia menjadi Algoritma.

Dalam pemrograman komputer, seringkali ada banyak cara berbeda - algoritma - untuk menyelesaikan tugas yang diberikan. Setiap algoritma memiliki kelebihan dan kekurangan dalam situasi yang berbeda. Penyortiran adalah satu tempat di mana banyak penelitian telah dilakukan karena komputer menghabiskan banyak daftar penyortiran waktu. Berikut adalah lima algoritma berbeda yang digunakan dalam penyortiran:

1. Bin sort
2. Gabungkan semacam
3. Semacam gelembung
4. Semacam shell
5. Quicksort

Jika ada sejuta nilai integer antara 1 dan 10 dan perlu diurutkan, jenis bin sort adalah algoritma yang tepat untuk digunakan. Jika Anda memiliki sejuta judul buku, quicksort mungkin merupakan algoritma terbaik. Dengan mengetahui

kekuatan dan kelemahan dari berbagai algoritma, Anda memilih yang terbaik untuk tugas yang ada.

2.4.1 Desain Konseptual

Algoritma adalah serangkaian instruksi, sering disebut sebagai "proses," yang harus diikuti ketika memecahkan masalah tertentu. Meskipun secara teknis tidak dibatasi oleh definisi, kata itu hampir selalu terkait dengan komputer, karena algoritma yang diproses komputer dapat mengatasi masalah yang jauh lebih besar daripada manusia, jauh lebih cepat. Karena komputasi modern menggunakan algoritma jauh lebih sering daripada pada titik lain dalam sejarah manusia, bidang telah tumbuh di sekitar desain, analisis, dan penyempurnaan. Bidang desain algoritma membutuhkan latar belakang matematika yang kuat, dengan gelar ilmu komputer yang sangat dicari kualifikasi. Ini menawarkan semakin banyak pilihan karir yang sangat dikompensasi, karena kebutuhan akan lebih banyak (dan juga lebih canggih) algoritma terus meningkat.

Pada tingkat yang paling sederhana, algoritma pada dasarnya hanya seperangkat instruksi yang diperlukan untuk menyelesaikan tugas. Pengembangan algoritma, meskipun umumnya tidak disebut demikian, telah menjadi kebiasaan yang populer dan pengejaran profesional untuk semua catatan sejarah. Jauh sebelum fajar era komputer modern, orang menetapkan rutinitas yang telah ditentukan untuk bagaimana mereka akan melakukan tugas sehari-hari, sering menuliskan daftar langkah-langkah yang harus diambil untuk mencapai tujuan penting, mengurangi risiko melupakan sesuatu yang penting. Ini, pada dasarnya, adalah apa itu algoritma. Desainer mengambil pendekatan yang mirip dengan

pengembangan algoritma untuk tujuan komputasi: pertama, mereka melihat masalah. Kemudian, mereka menguraikan langkah-langkah yang akan diperlukan untuk menyelesaikannya. Akhirnya, mereka mengembangkan serangkaian operasi matematika untuk mencapai langkah-langkah tersebut.

2.4.2 Tugas Algoritma

Tugas sederhana dapat diselesaikan dengan algoritma yang dihasilkan dengan beberapa menit, atau paling banyak pekerjaan pagi. Tingkat kompleksitas menjalankan tantangan yang panjang, namun, sampai pada masalah yang sangat rumit sehingga mereka telah menghalangi matematikawan yang tak terhitung jumlahnya selama bertahun-tahun - atau bahkan berabad-abad. Komputer modern menghadapi masalah pada tingkat ini di bidang-bidang seperti keamanan dunia maya, serta penanganan data besar - penyortiran set data yang efisien dan menyeluruh sedemikian besar sehingga bahkan komputer standar tidak dapat memprosesnya secara tepat waktu. Contoh data besar mungkin termasuk "setiap artikel di Wikipedia," "setiap halaman web yang diindeks dan diarsipkan akan kembali ke tahun 1998," atau "enam bulan terakhir pembelian online yang dilakukan di Amerika."

2.4.3 Rekayasa Algoritma

Ketika desain algoritma baru diterapkan dalam istilah praktis, disiplin terkait dikenal sebagai rekayasa algoritma. Kedua fungsi tersebut sering dilakukan oleh orang yang sama, meskipun organisasi yang lebih besar (seperti Amazon dan Google) mempekerjakan desainer dan insinyur khusus, mengingat tingkat kebutuhan mereka akan algoritma baru dan khusus. Seperti proses desain, rekayasa

algoritma sering kali melibatkan akreditasi sains komputer, dengan latar belakang yang kuat dalam matematika: di mana mereka ada sebagai profesi yang terpisah dan terspesialisasi, insinyur algoritma mengambil ide-ide konseptual dari desainer dan proses kreatif dari mereka yang akan dipahami oleh komputer. Dengan kemajuan teknologi digital yang mantap, para insinyur yang berdedikasi akan terus menjadi semakin umum.

2.5 Steganografi

Kata steganografi (*steganography*) berasal dari bahasa Yunani yaitu *steganos* yang artinya tersembunyi atau terselubung dan *graphein*, yang artinya menulis, sehingga kurang lebih artinya adalah menulis tulisan yang tersembunyi atau terselubung (Cokro, 2016). Secara umum steganografi merupakan seni atau ilmu yang digunakan untuk menyembunyikan pesan rahasia dengan segala cara sehingga selain orang yang dituju, orang lain tidak akan menyadari keberadaan dari pesan rahasia tersebut. Dari definisi diatas, maka dapat disimpulkan bahwa steganografi dibuat untuk membantu mengamankan informasi dengan cara menyembunyikan pesan pada media gambar, *audio*, ataupun *video*, agar pihak lain tidak mengetahui keberadaan informasi rahasia tersebut, kecuali si pengirim pesan dan penerima pesan. Dalam proses steganografi terdapat beberapa kriteria yang harus dipenuhi, kriterianya adalah sebagai berikut:

1. *Imperceptibility*

Keberadaan pesan tidak dapat dipersepsi oleh indra manusia, baik indra pendengaran maupun indra penglihatan.

2. *Fidelity*

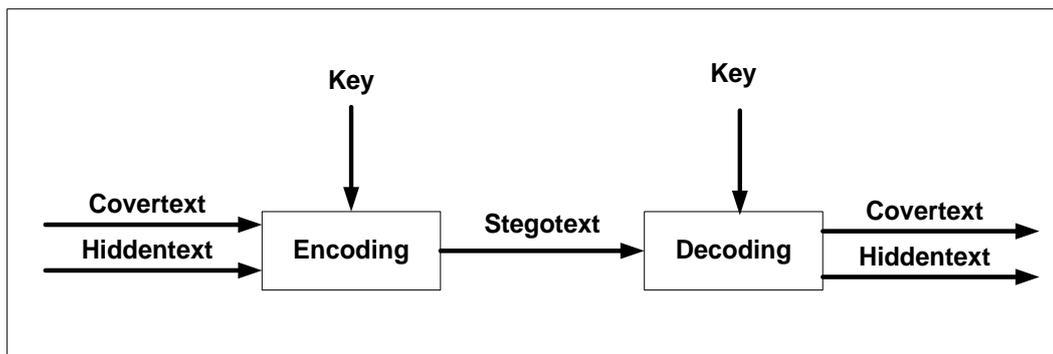
Mutu dari citra penampung tidak jauh berubah. Setelah penambahan pesan rahasia, citra hasil *steganografi* masih terlihat dengan baik. Pengamat tidak mengetahui kalau di dalam citra terdapat pesan rahasia

3. *Recovery*

Pesan rahasia yang disembunyikan di dalam citra digital harus dapat diungkapkan kembali seperti aslinya. Terdapat beberapa istilah yang berkaitan dengan *steganografi*:

- a. *Hiddentext* atau *embedded message*, pesan yang disembunyikan.
- b. *Coverttext* atau *cover-object*, pesan yang digunakan untuk menyembunyikan *embedded message*.
- c. *Stegotext* atau *stego-object*, pesan yang sudah berisi *embedded message*.

Steganografi menggunakan media gambar ini, *hidden text* atau *embedded message* yang dimaksudkan adalah teks yang akan disisipkan ke dalam *coverttext* atau *coverobject* yaitu *file* gambar yang digunakan sebagai media penampung pesan yang akan disisipkan. Dari hasil *encoding* atau *embedding* Pesan kedalam *file* gambar akan dihasilkan *stegotext* atau *stego-object* yang merupakan *file* gambar yang berisikan pesan *embedding*.



Gambar 2.1 Skema Steganografi

Penyisipan pesan ke dalam media *coverttext* dinamakan *encoding*, sedangkan ekstraksi pesan dari *stegotext* dinamakan *decoding*. Kedua proses ini memerlukan kunci rahasia (*stegokey*) agar hanya pihak yang berhak saja yang dapat melakukan penyisipan dan ekstraksi pesan, seperti yang terlihat pada gambar 2.1. Pada dasarnya, terdapat tujuh teknik yang digunakan dalam steganografi:

1. *Injection*

Penanaman adalah suatu teknik menanamkan pesan rahasia secara langsung ke suatu media. Salah satu masalah dari teknik ini adalah ukuran media yang diinjeksi menjadi lebih besar dari ukuran normalnya sehingga mudah dideteksi. Teknik itu sering juga disebut Embedding.

2. Substitusi

Substitusi data normal digantikan dengan data rahasia. Biasanya, hasil teknik itu tidak terlalu mengubah ukuran data asli, tetapi tergantung pada file media dan data yang akan disembunyikan. Teknik substitusi bisa menurunkan kualitas media yang ditumpangi.

3. *Transform Domain*

Teknik ini sangat efektif. Pada dasarnya, transformasi domain menyembunyikan data pada “*transform space*”.

4. *Spread Spectrum*

Sebuah teknik pentransmisiian menggunakan *pseudo-noise code*, yang independen terhadap data informasi sebagai modulator bentuk gelombang untuk menyebarkan energy sinyal dalam sebuah jalur komunikasi (*bandwidth*) yang lebih besar daripada sinyal jalur komunikasi informasi. Oleh penerima, sinyal dikumpulkan kembali menggunakan replika pseudo-noise code tersinkronisasi.

5. *Statistical Method*

Teknik ini disebut juga skema steganographic 1-bit. Skema tersebut menanamkan 1-bit informasi pada media tumpangan dan mengubah statistic walaupun hanya 1-bit. Perubahan statistik ditunjukkan dengan indikasi 1 dan jika tidak ada perubahan, terlihat indikasi 0. Sistem ini bekerja berdasarkan kemampuan penerima dalam membedakan antara informasi yang dimodifikasi dan yang belum.

6. *Distortion*

Metode ini menciptakan perubahan atas benda yang ditumpangi oleh data rahasia

2.5.1 Sejarah Steganografi

Steganografi dengan media kepala budak (dikisahkan oleh Herodatus, penguasa Yunani pada tahun 440 BC di dalam buku: *Histories of Herodatus*). Kepala budak dibotaki, ditulisi pesan, rambut budak dibiarkan tumbuh, budak dikirim. Di tempat penerima kepala budak digunduli agar pesan bisa dibaca. Penggunaan tinta tak-tampak (*invisible ink*). Tinta dibuat dari campuran sari buah, susu, dan cuka. Tulisan di atas kertas dapat dibaca dengan cara memanaskan kertas tersebut (Pratiwi, 2017).

2.5.2 Stego File

Stego File adalah media dimana informasi akan disisipkan. Media ini dapat berupa file gambar, video, suara atau jenis file apapun. Teknik menyimpan informasi pada file ini berbeda-beda tergantung algoritma yang digunakan.

2.6 Audio

Audio adalah suara atau bunyi yang dihasilkan ketika molekul udara berubah oleh suatu gerakan yang ditimbulkan sebuah objek yang menghasilkan sebuah getaran. Objek ini bias berupa senar gitar, vokal manusia, atau sebuah kaleng yang bergerak ketika ada energi yang menggetarkannya. Ketika program memetik senar gitar pada sebuah gitar, senar tersebut akan bergerak maju dan mundur dengan jumlah getaran tertentu. Jumlah getaran ini disebut sebagai frekuensi dari getaran tersebut. Satu kali gerakan maju dan mundur tersebut disebut *cycle* (putaran). Maka satuan untuk frekuensi adalah *Cycle Per Second*, atau CPS. Satuan ini biasa dikenal dengan sebutan Hertz (Hz). Kadangkala getaran yang

dihasilkan sangat cepat, dan perbandingan gerakan yang dihasilkan adalah seribu, maka satuan yang digunakan adalah kilohertz (KHz).

Pada umumnya format audio dibagi menjadi dua yaitu audio analog dan audio digital. Berikut penjelasan lengkapnya:

1. Audio Analog

Audio analog adalah harmonisasi bunyi yang hanya bisa dibuat melalui suara sintetis yang disimpan dalam bentuk media kaset pita. Format analog yang hanya dapat menyimpan suara dalam jumlah yang terbatas.

2. Audio Digital

Audio digital adalah harmonisasi bunyi yang dibuat melalui perekaman konvensional maupun suara sintetis yang disimpan dalam media berbasis teknologi komputer. Format digital dapat menyimpan data dalam jumlah besar, jangka panjang dan berjaringan luas. Audio digital menggunakan sinyal digital dalam proses reproduksi suaranya. Sebagai proses digitalisasi terhadap format rekaman musik analog, lagu atau musik digital mempunyai beraneka ragam format yang bergantung pada teknologi yang digunakan.

Format-format audio digital yang digunakan pada penelitian ini adalah sebagai berikut:

1. WAV

File wave (WAV) adalah format file audio yang diciptakan oleh microsoft, dan telah menjadi standar format audio bagi PC. File WAV dikenali sebagai sebuah file dengan ekstensi *.wav. File WAV memiliki format 1- N saluran

dan 16 bit per sample, serta memiliki sampling rate 44.1 KHz. File WAV digunakan di dalam PC sebagai suatu medium pertukaran antara komputer dengan platform yang berbeda. Sound sample juga disimpan sebagai data mentah dengan 1-N saluran dalam file yang sama. Saluran tersebut harus diselipkan dengan cara yang sama seperti dalam file AIFF-C. Sebagai tambahan untuk data mentah audio yang tidak terkompresi, format file WAV menyimpan informasi mengenai jumlah track (mono atau stereo), sample rate, dan bit depth. Keuntungan dengan adanya file WAV adalah kita dapat mengedit file tersebut karena file tersebut masih dalam kondisi tidak terkompresi, tetapi ketika file telah dimampatkan, kita akan mengalami kesulitan dalam mengedit.

2. MP3

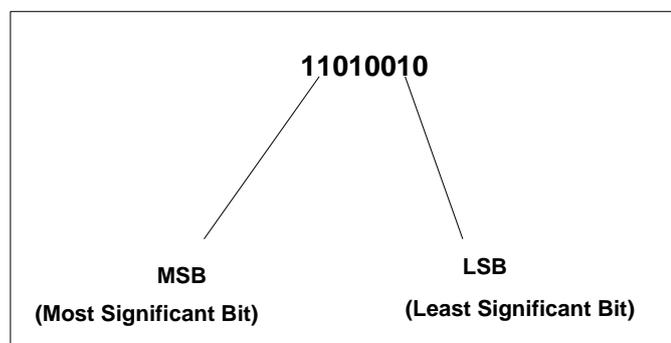
MP3 (MPEG, Audio Layer 3) menjadi format paling populer dalam musik digital. Hal ini dikarenakan ukuran filenya yang kecil dengan kualitas yang tidak kalah dengan CD Audio. *Format* ini dikembangkan dan dipatenkan oleh Fraunhofer Institute. Dengan Bitrate 128 Kbps, file MP3 sudah berkualitas baik. Namun MP3 Pro-format penerus MP3 menawarkan kualitas yang sama dengan Bitrate setengah dari MP3. MP3 Pro-kompatibel dengan MP3. Pemutar MP3 dapat memainkan file MP3 Pro namun kualitas suaranya tidak sebagus pengganti yang mendukung MP3 Pro. MP3 terdiri dari bagian-bagian kecil yang disebut *frame*. Setiap *frame* memiliki waktu yang konstan yaitu 0,026 detik (26 milidetik). Tapi ukuran dari sebuah *frame* (dalam *byte*) bervariasi bergantung pada bitrate. Sebuah *frame* MP3

secara umum terdiri atas 5 bagian, yaitu *Header*, *CRC*, *Side Information*, *Main Data* dan *Ancillary Data*.

2.7 Least Significant Bit

Salah satu metode yang digunakan dalam teknik steganografi pada tipe berkas *audio* dan gambar adalah least significant bit atau disebut juga *low bit encoding*. Metode ini berasal dari angka yang paling kurang signifikan dari jumlah bit dalam 1 *byte*. Bit yang memiliki signifikansi paling tinggi adalah numerik yang memiliki nilai tertinggi (misal, $2^7 = 128$), artinya yang paling tidak signifikan adalah yang memiliki nilai terendah (misal, $2^0 = 1$) (Girsang, 2017).

Penyembunyian pesan dilakukan dengan menggantikan *bit-bit* didalam segmen citra dengan *bit-bit* pesan rahasia. Metode yang paling sering digunakan adalah metode modifikasi *LSB (Least Significant Bit)* pada citra penampung. Pada susunan bit di dalam sebuah *byte* (1 *byte* = 8 bit), ada bit paling signifikan yang disebut *MSB (Most Significant Bit)* dan bit yang paling kurang signifikan atau *LSB (Least Significant Bit)*.



Gambar 2.2 Contoh susunan bit pada *LSB* dan *MSB*

Sumber: (Segi3Hijau, 2012)

Contoh susunan bit pada byte yang menjelaskan bit yang cocok untuk diganti adalah bit LSB, sebab penggantian hanya mengubah nilai byte satu lebih tinggi atau satu lebih rendah dari nilaisebelumnya. Misalkan byte di dalam gambar menyatakan warna tertentu, maka perubahan pada bit LSBnya tidak mengubah warna secara signifikan. Sebelum melakukan penggantian bit-bit LSB, semua data citra yang tidak bertipe 24-bit diubah terlebih dahulu menjadi format 24-bit. Jadi setiap data piksel sudah mengandung komponen warna merah, hijau dan biru (RGB). Nilai dari bit-bit yang kurang signifikan atau LSB dari setiap byte di dalam bitmap digantikan dengan bit-bit pesan yang akan disembunyikan. Jika byte merupakan komponen hijau (G), maka penggantian satu bit LSB-nya hanya mengubah sedikit kadar warna hijau, dan perubahannya tak terdeteksi oleh mata manusia.

Contoh penggunaan Metode LSB pada tahapan encode:

Misalkan penyisipan pada citra 24-bit. Setiap pixel panjangnya 24 bit (3 x 3 byte, masing-masing komponen R (1 byte), G (1 byte), B (1 byte)).

00110011 10100010 11100010 (misal pixel warna merah)

Misalkan embedded message: 010

Encoding 00110010 1010001 1 11100010

(pixel warna 'merah berubah sedikit'. Tidak dapat dibedakan secara visual dengan citra aslinya)

Jika pesan = 10 bit, maka jumlah byte yang digunakan = 10 byte

00110011 10100010 11100010

10101011 00100110

10010110 11001001 11111001

10001000 10100011

Pesan:

1110010111

Hasil penyisipan pada bit LSB:

0011001

1 1010001

11110001

1 1010101

00010011

01001011

11100100

011111001

1000100

110100011

Berkas carrier dalam biner dengan ukuran 8 byte :

10010101 00001101 11001001 10010110 00001111 11001011 10011111

00010000.

Karakter 'G' dalam biner dengan ukuran 1 byte : 01000111

Proses Least Significant Bit Modification:

1001010000001101 11001000 10010110 0000111011001011 10011111 00010001

2.8 Unified Modelling Language (UML)

Unified Modeling Language adalah salah satu perkakas (*tool*) yang sangat bermanfaat untuk melakukan analisis dan perancangan sistem dalam konteks “pemrograman berorientasi objek” perangkat lunak yang berparadigma berorientasi objek. Pemodelan (*modeling*) sesungguhnya digunakan untuk penyederhanaan permasalahan-permasalahan yang kompleks sedemikian rupa sehingga lebih mudah dipelajari dan dipahami”.

Penggunaan model ini bertujuan untuk mengidentifikasi bagian-bagian yang termasuk dalam lingkup sistem yang dibahas dan bagaimana hubungan antara sistem dengan subsistem maupun sistem lain diluarnya(Wasserkrug et al., 2009).

Penggunaan model ini bertujuan untuk mengidentifikasi bagian-bagian yang termasuk dalam lingkup sistem yang dibahas dan bagaimana hubungan antara sistem dengan subsistem maupun sistem lain diluarnya (Sukmawati & Priyadi, 2019).

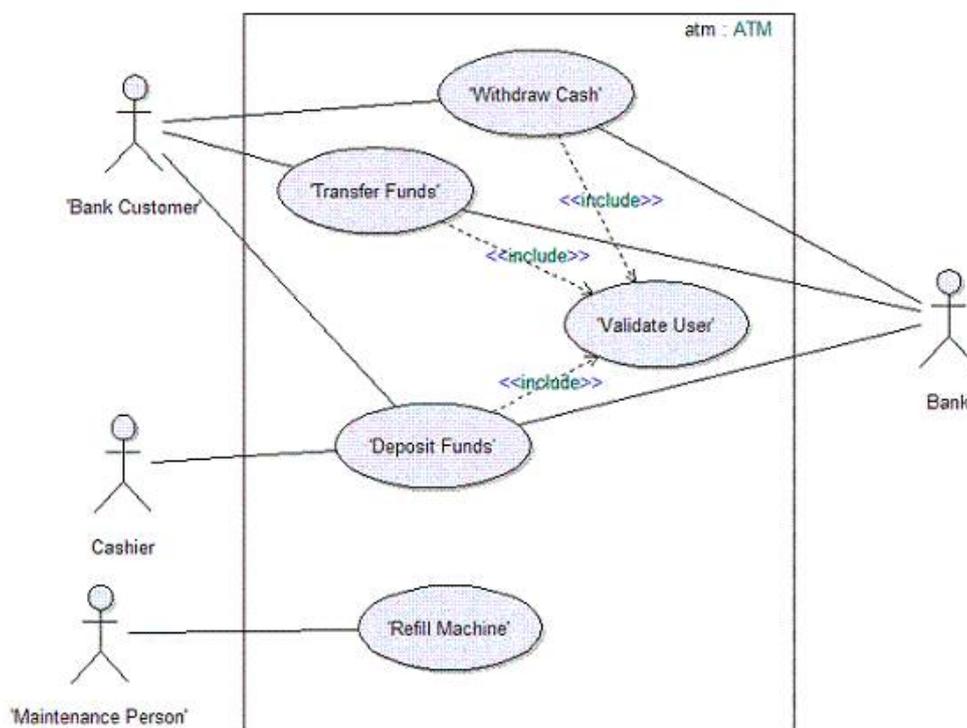
2.8.1 Use Case Diagram

Use Case Diagram adalah model tentang bagaimana berbagai jenis pengguna berinteraksi dengan sistem untuk memecahkan masalah. Dengan demikian, ini menggambarkan tujuan pengguna, interaksi antara pengguna dan

sistem, dan perilaku sistem yang diperlukan dalam memenuhi tujuan-tujuan ini. Model use-case terdiri dari sejumlah elemen model. Elemen model yang paling penting adalah kasus penggunaan, aktor dan hubungan di antara mereka. Diagram use-case digunakan untuk menggambarkan secara grafis subset dari model untuk menyederhanakan komunikasi. Biasanya akan ada beberapa diagram kasus penggunaan yang terkait dengan model yang diberikan, masing-masing menunjukkan subset elemen model yang relevan untuk tujuan tertentu. Elemen model yang sama dapat ditampilkan pada beberapa diagram use-case, tetapi setiap instance harus konsisten. Jika alat digunakan untuk mempertahankan model use-case, kendala konsistensi ini otomatis sehingga setiap perubahan pada elemen model (mengubah nama misalnya) akan secara otomatis tercermin dalam setiap diagram use-case yang menunjukkan elemen itu.

Model use-case dapat berisi paket yang digunakan untuk menyusun model untuk menyederhanakan analisis, komunikasi, navigasi, pengembangan, pemeliharaan, dan perencanaan. Faktanya, sebagian besar model use-case adalah tekstual, dengan teks yang ditangkap dalam Spesifikasi Use-Case yang terkait dengan setiap elemen model use-case. Spesifikasi ini menjelaskan alur peristiwa use case. Model use-case berfungsi sebagai utas pemersatu sepanjang pengembangan sistem. Ini digunakan sebagai spesifikasi utama dari persyaratan fungsional untuk sistem, sebagai dasar untuk analisis dan desain, sebagai input untuk perencanaan iterasi, sebagai dasar mendefinisikan kasus uji dan sebagai dasar untuk dokumentasi pengguna. (Kurniawan, 2018).

Use case diagram merupakan suatu diagram yang berisi *use case*, *actor*, serta *relationship* diantaranya. *Use Case Diagram* dapat digunakan untuk kebutuhan apa saja yang diperlukan dalam suatu sistem, sehingga sistem dapat digambarkan dengan jelas bagaimana proses dari sistem tersebut, bagaimana cara aktor menggunakan sistem, serta apa saja yang dapat dilakukan pada suatu sistem.

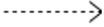


Gambar 2.3 Use-case Diagram ATM

Sumber: (Hendini., 2016)

Gambar di atas adalah contoh dari penggunaan use-case diagram pada mesin ATM. Use-case memiliki beberapa simbol untuk menyatakan kegiatan dari use-case tersebut. Adapun simbol dari *use case* adalah sebagai berikut:

Tabel 2.1 Simbol Use Case Diagram

| No | Gambar | Nama | Keterangan |
|----|---|-----------------------|---|
| 1 |  | <i>Actor</i> | Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> . |
| 2 |  | <i>Dependency</i> | Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri. |
| 3 |  | <i>Generalization</i> | Hubungan dimana objek anak berbagi perilaku dan struktur data dari objek yang ada di atasnya. |
| 4 |  | <i>Include</i> | Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> . |
| 5 |  | <i>Extend</i> | Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan. |
| 6 |  | <i>Association</i> | Apa yang menghubungkan antara objek satu dengan objek lainnya. |

| | | | |
|----|---|----------------------|---|
| 7 |  | <i>System</i> | Menspesifikasikan paket yang menampilkan sistem secara terbatas. |
| 8 |  | <i>Use Case</i> | Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor |
| 9 |  | <i>Collaboration</i> | Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi). |
| 10 |  | <i>Note</i> | Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi |

Sumber: (Kurniawan, 2018)

2.8.2 Activity Diagram

Activity Diagram (Diagram Aktifitas) menggambarkan berbagai alir aktifitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir (Ladjamudin, 2005).

Activity diagram menurut adalah salah satu cara untuk memodelkan *event-event* yang terjadi dalam suatu *use case*. Diagram ini juga dapat digantikan dengan sejumlah teks.

Tabel 2.2 Simbol Activity Diagram

| No | Gambar | Nama | Keterangan |
|----|---|--------------------------------------|---|
| 1 |  | <i>Activity</i> | Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain |
| 2 |  | <i>Action</i> | State dari sistem yang mencerminkan eksekusi dari suatu aksi |
| 3 |  | <i>Initial Node</i> | Bagaimana objek dibentuk /diawali. |
| 4 |  | <i>Activity</i> <i>Final Node</i> | Bagaimana objek dibentuk dan dihancurkan |
| 5 |  | <i>Fork Node</i> | Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran |

Sumber: (Kurniawan, 2018)

2.9 Flowchart

Flowchart digunakan dalam mendesain dan mendokumentasikan proses atau program sederhana. Seperti jenis diagram lainnya, diagram membantu memvisualisasikan apa yang sedang terjadi dan dengan demikian membantu

memahami suatu proses, dan mungkin juga menemukan fitur-fitur yang kurang jelas dalam proses tersebut, seperti kekurangan dan hambatan. Ada berbagai jenis diagram alur: masing-masing jenis memiliki set kotak dan notasi sendiri. Dua jenis kotak yang paling umum dalam diagram alur adalah:

1. langkah pemrosesan, biasanya disebut aktivitas dan dilambangkan sebagai kotak persegi panjang.
2. keputusan biasanya dilambangkan sebagai berlian.

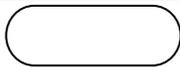
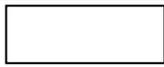
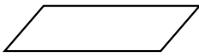
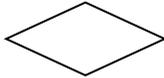
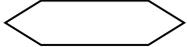
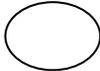
Diagram alir digambarkan sebagai "lintas fungsional" ketika bagan dibagi menjadi bagian vertikal atau horizontal yang berbeda, untuk menggambarkan kontrol unit organisasi yang berbeda. Simbol yang muncul di bagian tertentu berada dalam kendali unit organisasi itu. Flowchart lintas fungsional memungkinkan penulis untuk menemukan tanggung jawab untuk melakukan suatu tindakan atau membuat keputusan dengan benar, dan untuk menunjukkan tanggung jawab masing-masing unit organisasi untuk bagian berbeda dari satu proses tunggal.

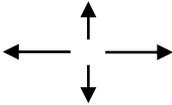
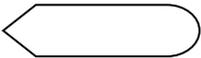
Diagram alir menggambarkan aspek-aspek tertentu dari proses dan biasanya dilengkapi dengan jenis diagram lainnya. Misalnya, Kaoru Ishikawa, mendefinisikan diagram alir sebagai salah satu dari tujuh alat dasar kendali mutu, di sebelah histogram, diagram Pareto, lembar periksa, diagram kontrol, diagram sebab-akibat, dan diagram sebaran. Demikian pula, di UML, notasi pemodelan konsep standar yang digunakan dalam pengembangan perangkat lunak, diagram aktivitas, yang merupakan jenis diagram alur, hanyalah salah satu dari banyak jenis diagram yang berbeda.

Diagram Nassi-Shneiderman dan Drakon-chart adalah notasi alternatif untuk aliran proses. Nama alternatif umum termasuk diagram alir, diagram alur proses, diagram alur fungsional, peta proses, diagram proses, diagram proses fungsional, model proses bisnis, model proses, diagram alir proses, diagram alur kerja, diagram alir bisnis. Istilah "diagram alur" dan "diagram alir" digunakan secara bergantian (Nakatsu, 2009).

Struktur grafik yang mendasari diagram alur adalah grafik aliran, yang mengabstraksi jenis simpul, isinya, dan informasi tambahan lainnya. Adapun simbol-simbol flowchart lihat pada tabel sebagai berikut :

Tabel 2.3 Simbol Flowchart

| NO | SIMBOL | FUNGSI |
|----|---|---|
| 1. |  | <i>Terminal</i> , untuk memulai atau mengakhiri suatu program |
| 2. |  | <i>Process</i> , suatu simbol yang menunjukkan setiap pengolahan yang dilakukan. |
| 3. |  | <i>Input-Output</i> , untuk memasukkan menunjukkan hasil dari suatu proses |
| 4. |  | <i>Decision</i> , suatu kondisi yang akan menghasilkan beberapa kemungkinan jawaban atau pilihan |
| 5. |  | <i>Preparation</i> , suatu symbol yang menyediakan tempat pengolahan |
| 6. |  | <i>Connector</i> , suatu prosedur penghubung yang akan masuk atau keluar melalui symbol ini dalam |

| | | |
|-----|---|--|
| | | lembar yang sama |
| 7. |  | <i>Off-Page Connector</i> , merupakan symbol masuk atau keluarannya suatu prosedur pada lembaran kertas lainnya |
| 8. |  | <i>Arus/Flow</i> , dari pada prosedur yang dapat dilakukan atas ke bawah dari bawah ke atas, ke atas dari kiri ke kanan ataupun dari kanan ke kiri |
| 9. |  | <i>Predefined Process</i> , untuk menyatakan sekumpulan langkah proses yang ditulis sebagai prosedur |
| 10. |  | Simbol untuk output, yang ditunjukkan ke suatu device, seperti printer, dan sebagainya |
| 11 |  | Penyimpanan file secara sementara |
| 12 |  | Menunjukkan <i>Input/Output Hardisk</i> (media penyimpanan) |

Sumber: (Kurniawan, 2018)

2.10 Visual Basic

Bahasa Pemrograman *Microsoft Visual Basic .NET* adalah sebuah bahasa pemrograman tingkat tinggi untuk *Microsoft .NET Framework*. Walaupun *VB.NET* ini memang dibuat supaya mudah dipahami dan dipelajari, namun bahasa pemrograman ini juga cukup *powerful* untuk memenuhi kebutuhan dari

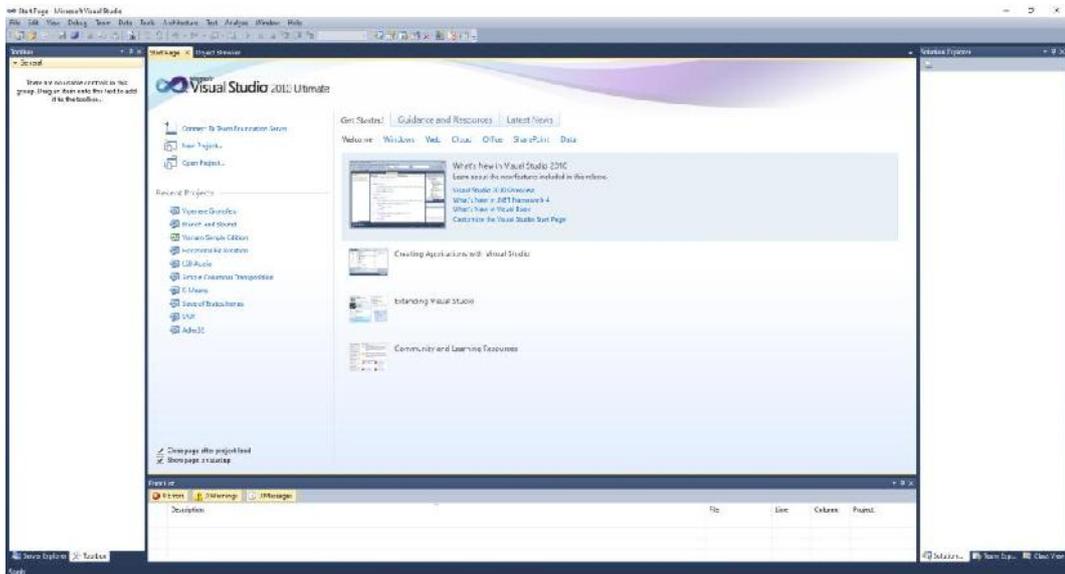
programmer yang berpengalaman. Bahasa pemrograman *Visual Basic .NET* mirip dengan bahasa pemrograman *Visual Basic*, namun keduanya tidak sama”.

Bahasa pemrograman *Visual Basic .NET* memiliki struktur penulisan yang mirip dengan bahasa Inggris, di mana hal ini juga menyebabkan kemudahan dalam membaca dan mengerti dari sebuah kode. Di mana dimungkinkan, kata ataupun frasa yang memiliki arti digunakan dan bukannya menggunakan singkatan, akronim ataupun *special characters*”.

Pada intinya *Visual Basic.NET* ini adalah sebuah bahasa pemrograman yang berorientasi pada *object*, yang bisa dianggap sebagai evolusi selanjutnya dari bahasa pemrograman *Visual Basic* standar (Wibowo, 2014).

2.10.1 Lingkungan kerja Visual Basic.Net

Pada saat pertama kali dijalankan Visual Basic 2010 Ultimate, akan menampilkan sebuah jendela Splash Visual Studio 2010 Ultimate, setelah jendela Splash Visual Studio 2010 Ultimate muncul kemudian akan keluar sebuah start page Microsoft Visual Studio seperti gambar 2.4.



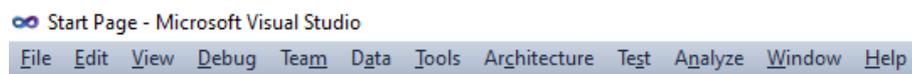
Gambar 2.4 Tampilan Microsoft Visual Studio 2010

2.10.2 Komponen Visual Basic.Net

Pada saat membuka program Visual Basic.Net, ada beberapa komponen yang terlihat. Berikut ini adalah beberapa komponen dari Visual Basic.Net:

1. Menu Bar

Menu Bar adalah bagian dari *IDE* yang terdiri atas perintah-perintah untuk mengatur *IDE*, mengedit kode, dan mengeksekusi program. Menu yang terdapat pada menu bar adalah *menu file, edit, view, project, build, debug, data, tools, window* dan *help*. *Menu bar* pada *Visual Studio 2010* seperti terlihat pada gambar 2.5.



Gambar 2.5 Tampilan Menu Bar

2. Toolbar

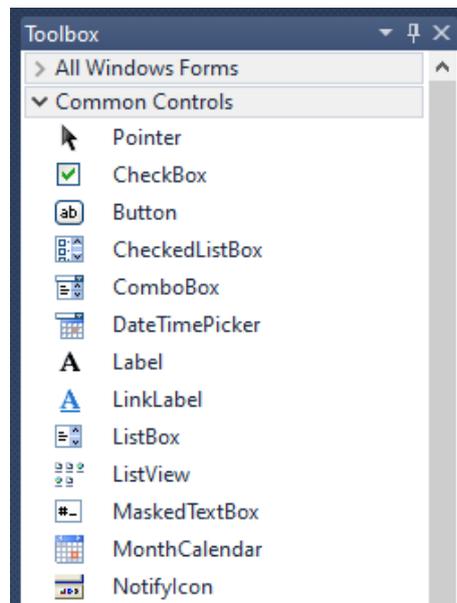
Fasilitas ini dapat mempercepat pengaksesan perintah-perintah yang ada dalam pemrograman seperti terlihat pada gambar 2.6.



Gambar 2.6 Tampilan Toolbar

3. Toolbox

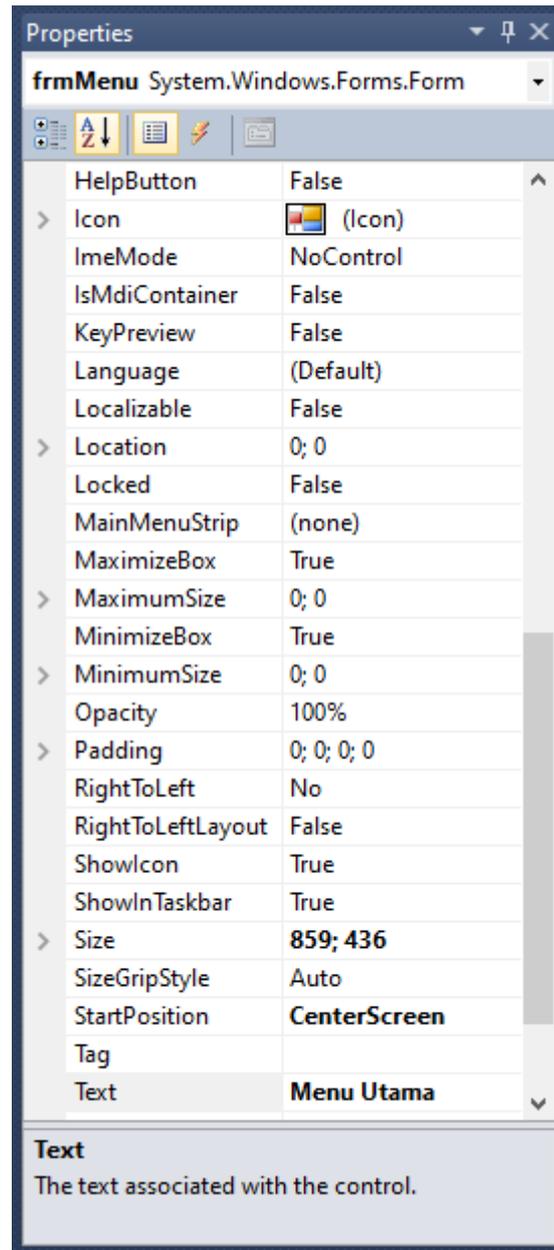
Sebuah *window* yang berisi tombol-tombol kontrol yang akan Anda gunakan untuk mendesain atau membangun sebuah *form* atau *report* seperti terlihat pada gambar 2.7.



Gambar 2.7 Tampilan Toolbox

4. Properties Window

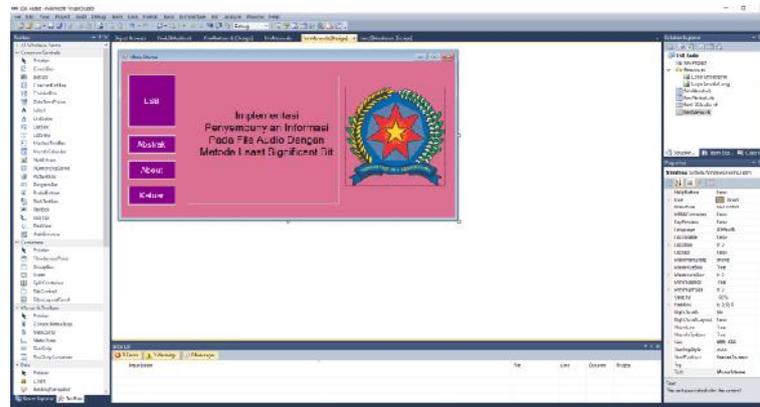
Properties window adalah tempat menyimpan *property* dari setiap objek control dan komponen.



Gambar 2.8 Tampilan Properties

5. Form

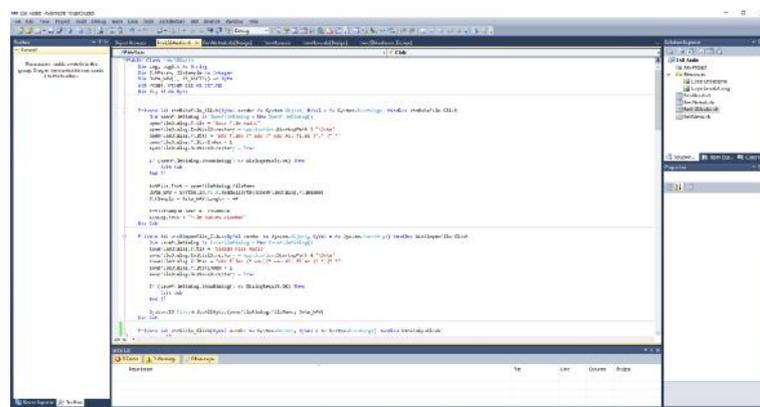
Form merupakan tempat di mana kontrol-kontrol diletakkan. Form juga berfungsi sebagai tempat pembuatan tampilan atau antarmuka (*user interface*) dari sebuah aplikasi *windows*.



Gambar 2.9 Tampilan Form

6. Code Editor

Code Editor adalah tempat di mana kita meletakkan atau menuliskan kode program dari program aplikasi kita.



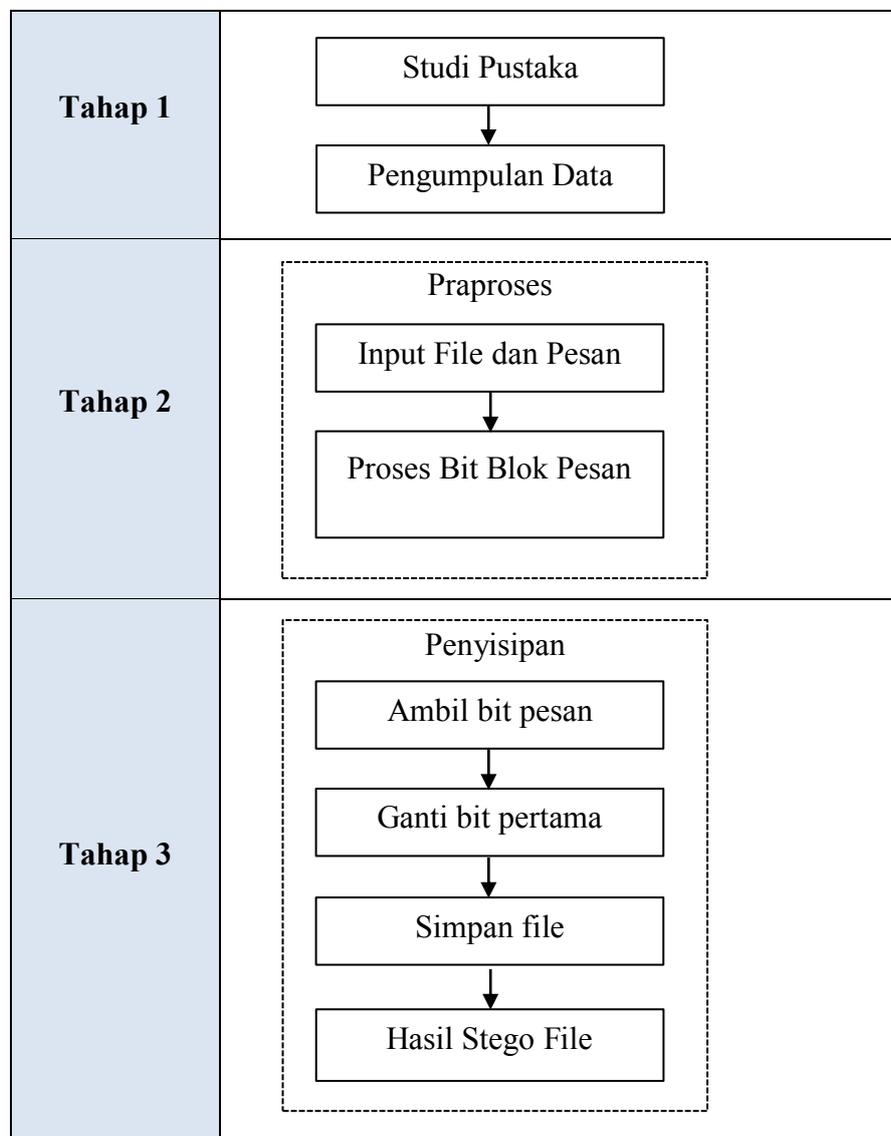
Gambar 2.10 Tampilan Code Editor

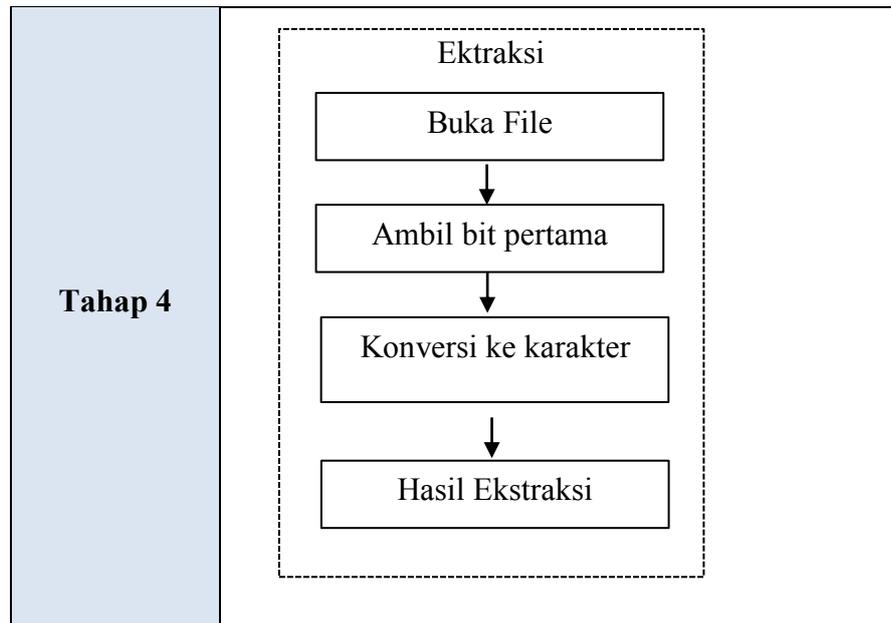
BAB III

METODE PENELITIAN

3.1. Kerangka Penelitian

Bagian ini akan membahas alur proses dari penyisipan teks pada file audio tersebut. Ada empat tahap yang dilakukan dalam proses percobaan program aplikasi yang akan dibuat. Gambar 3.1 adalah hasil dari kerangka penelitian.





Gambar 3.1 Kerangka Penelitian

Berikut adalah tahapan penelitian yang dilakukan:

- Studi Pustaka

Studi pustaka dilakukan untuk mendapatkan teori-teori tentang ilmu steganografi khususnya algoritma Least Significant Bit. Pembelajaran ini dilakukan dengan cara mencari teori dari buku dan informasi lainnya.

- Analisa

Bagian ini membuat penjelasan mengenai alur permasalahan dan solusi yang akan dilakukan untuk mencapai hasil. Proses analisa dilakukan untuk memastikan kebenaran dari rancangan yang akan digunakan pada program aplikasi.

- **Pembahasan**

Bagian ini membahas tentang metode yang digunakan oleh penulis tentang algoritma Least Significant Bit dalam melakukan proses penyisipan informasi dalam media suara.

- **Implementasi dan pengujian**

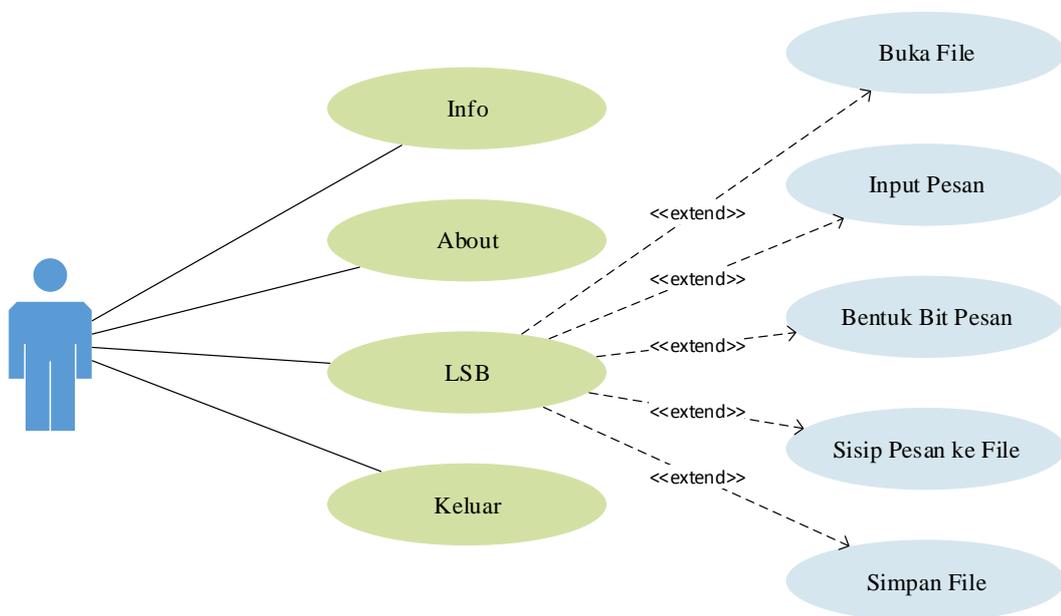
Bagian ini dilakukan pengujian kebenaran hasil yang dijalankan oleh program aplikasi steganografi yang dibuat menggunakan Microsoft Visual Basic.Net 2010. Hasil akan disesuaikan dengan teori dan perancangan yang sudah dibuat sebelumnya.

3.2. Perancangan Penelitian

Penelitian ini dirancang dengan cara yang terstruktur dan terarah. Dalam setiap proses akan digambarkan dengan diagram. Penggunaan diagram UML sangat membantu perancangan penelitian ini agar menghasilkan program aplikasi yang baik dan tersusun. Perancangan ini membutuhkan struktur yang baik agar tidak berjalan diluar perkiraan. Hal ini bertujuan agar pembuatan program aplikasi menghasilkan output yang baik dan efisien. Desain penelitian berikut ini dipaparkan dalam bentuk diagram Unified Modelling Language (UML). Pada UML. Setiap alur bagian pada program aplikasi akan tergambarkan dengan baik. Perancangan ini diharapkan hanya memiliki kesalah kecil sehingga perbaikan dapat dilakukan dengan mudah.

3.2.1. Use Case Diagram

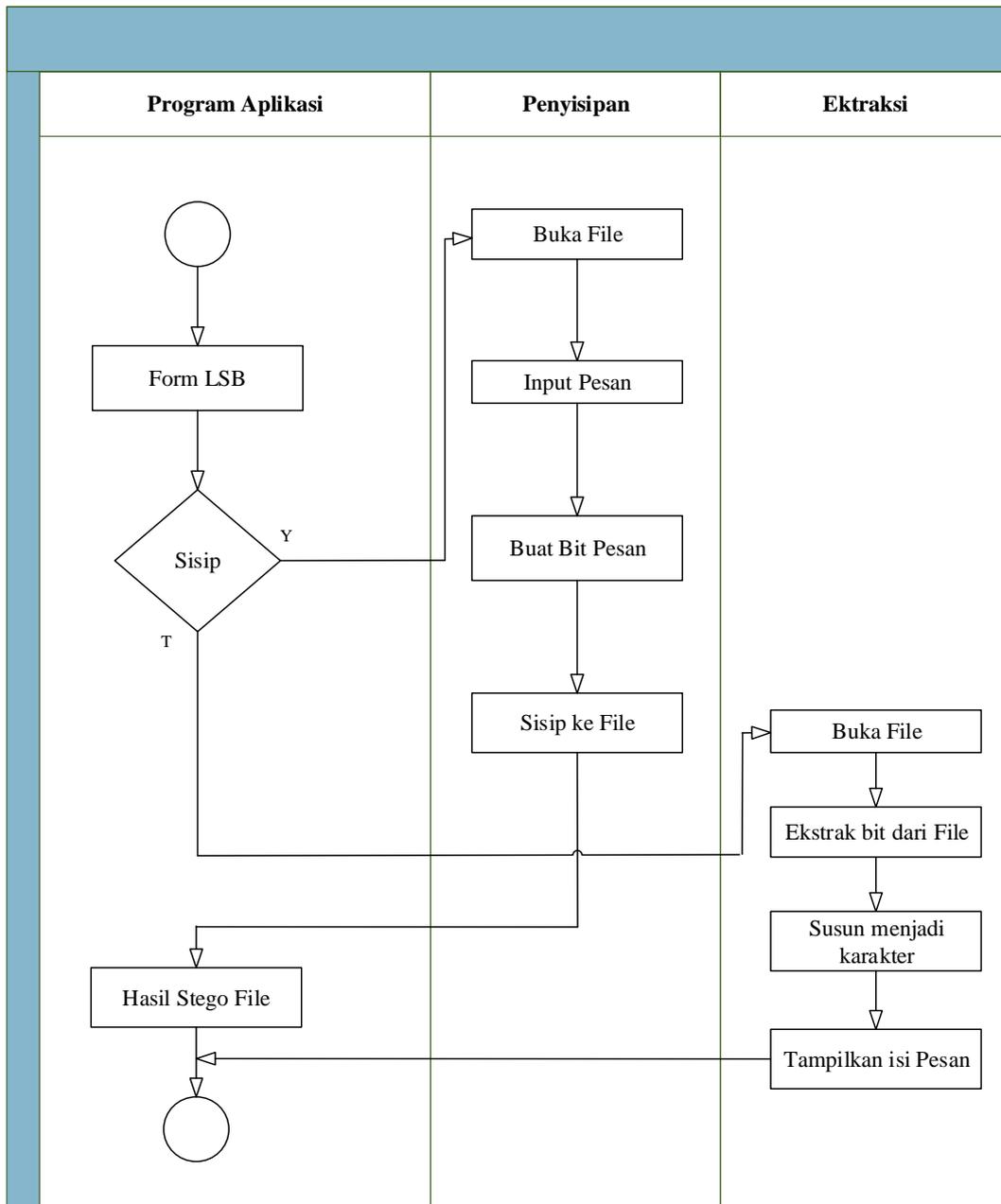
Use Case diagram didefinisikan sebagai diagram yang menentukan arah dari pengguna dan sistem. Use case diagram merupakan konsep utama dari penggambaran Unified Modeling Language. Gambar 3.2 adalah use case diagram yang digunakan dalam penelitian ini untuk mengetahui alur proses steganografi dengan metode LSB.



Gambar 3.2 Use Case Diagram

3.2.2. Activity Diagram

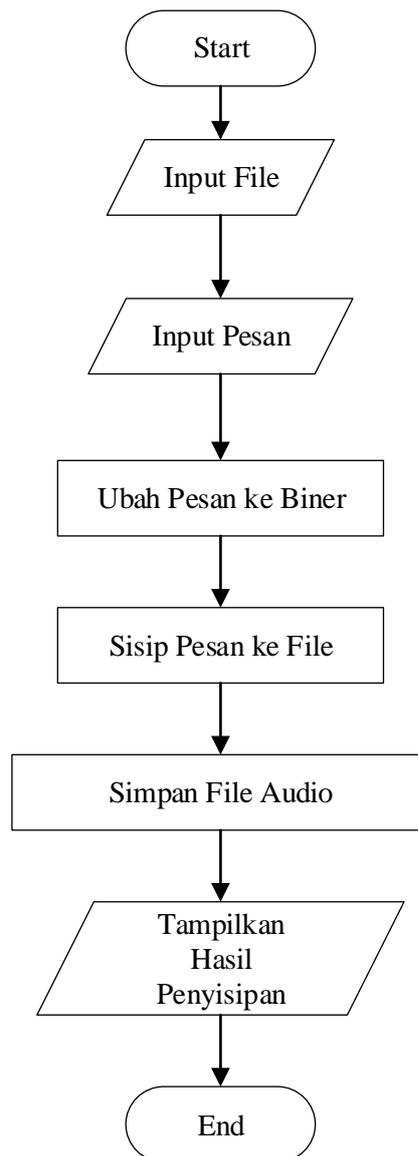
Activity diagram menggambarkan perilaku alur kerja aktual suatu sistem dalam Teknologi Informasi. Diagram ini menggambarkan keadaan aktual dari suatu sistem dengan menunjukkan semua urutan kegiatan yang dilakukan. Juga, diagram ini dapat menunjukkan aktivitas yang kondisional atau paralel. Gambar berikut ini akan menjelaskan *Activity diagram* tersebut.



Gambar 3.3 Activity Diagram penyisipan dan ekstraksi pesan

3.2.3. Flowchart Penyisipan

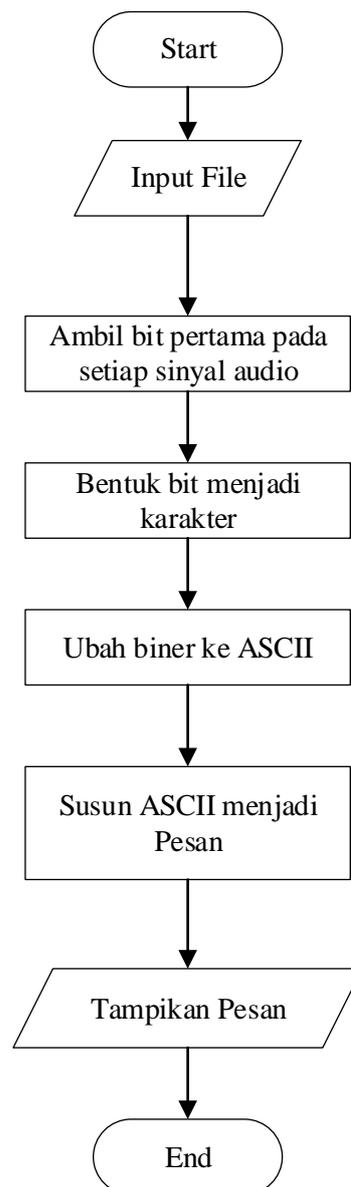
Flowchart penyisipan menjelaskan bagaimana suatu pesan akan tersimpan ke dalam file audio. *Flowchart* penyisipan pesan dapat dilihat pada gambar berikut ini.



Gambar 3.4 Flowchart penyisipan pesan

3.2.4. Flowchart Ekstraksi

Flowchart ekstraksi menjelaskan bagaimana suatu pesan akan diambil dari file audio yang sudah sebelumnya disimpan. *Flowchart* ekstraksi dapat dilihat pada gambar berikut ini.



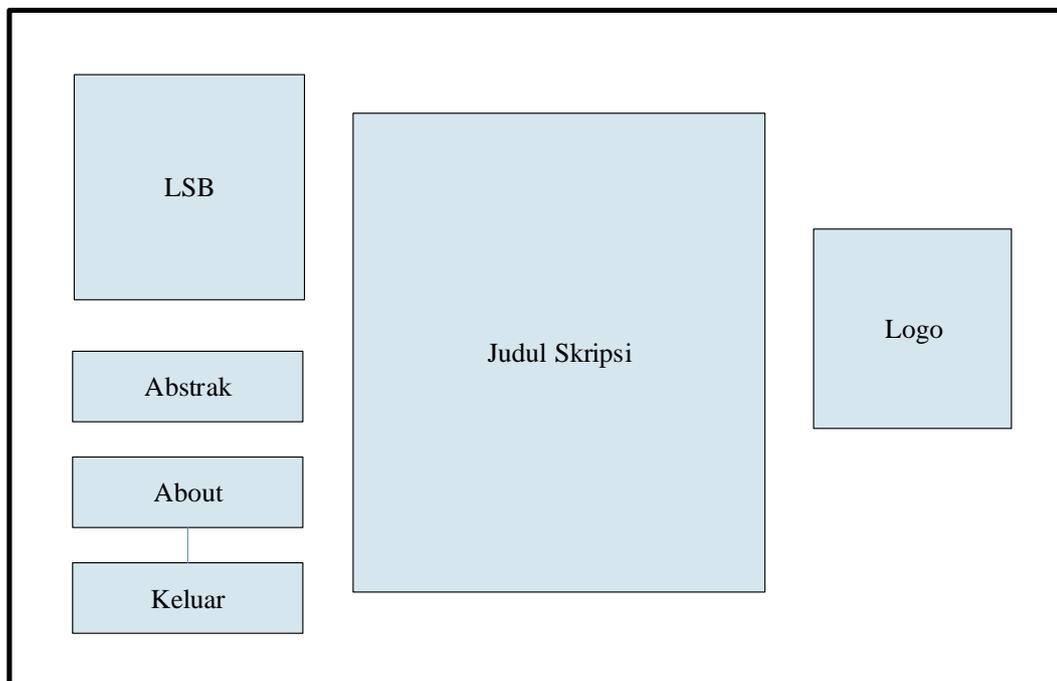
Gambar 3.5 Flowchart ekstraksi pesan

3.3. Desain Interface

Desain interface adalah perancangan tampilan program aplikasi akan muncul pada saat program dikerjakan. Perancangan ini melibatkan beberapa objek yang akan disematkan pada program aplikasi tersebut. Perancangan ini menampilkan komponen-komponen yang ada pada program aplikasi tersebut. Perancangan ini memudahkan penulis dalam pembuatan program aplikasi steganografi dengan metode LSB.

3.3.1. Menu Utama

Menu utama adalah tampilan yang pertama sekali muncul pada saat program aplikasi dijalankan. Gambar 3.6 adalah hasil perancangan menu utama yang memiliki beberapa komponen lainnya.



Gambar 3.6 Tampilan Menu Utama

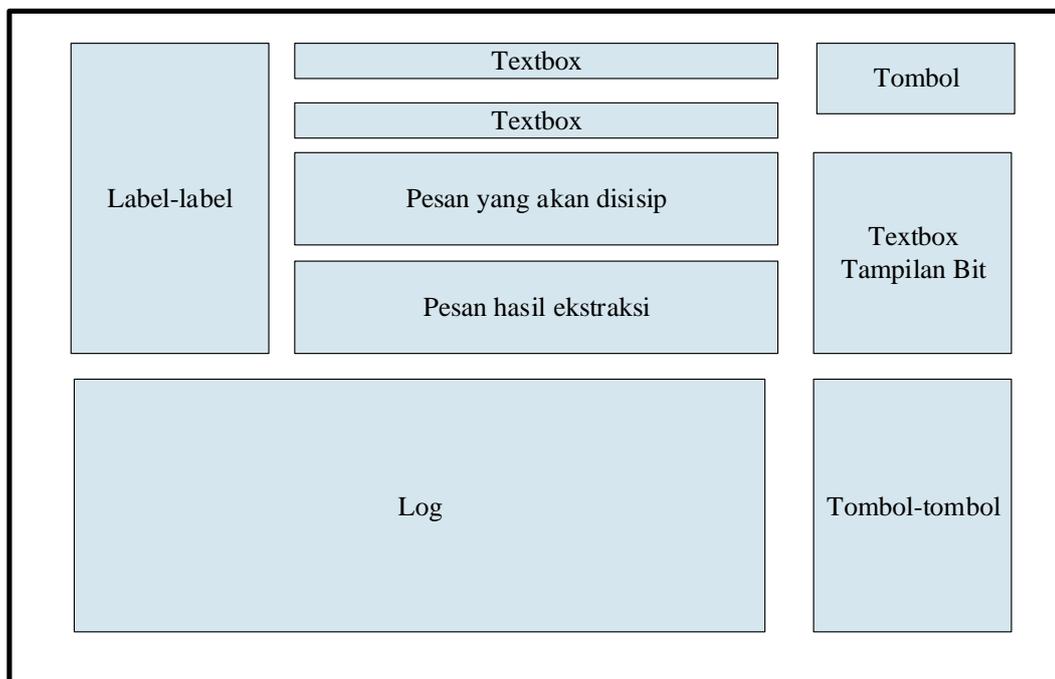
Tampilan ini memiliki berapa sub-menu antara lain:

- LSB
- Abstrak
- About
- Keluar
- Judul Skripsi
- Logo

3.3.2. Menu LSB

Menu ini adalah perancangan penyimpanan pesan dalam media audio.

Tampilan ini adalah tempat untuk melakukan proses steganografi dengan metode LSB. Gambar 3.7 adalah tampilan menu LSB.



Gambar 3.7 Tampilan Menu LSB

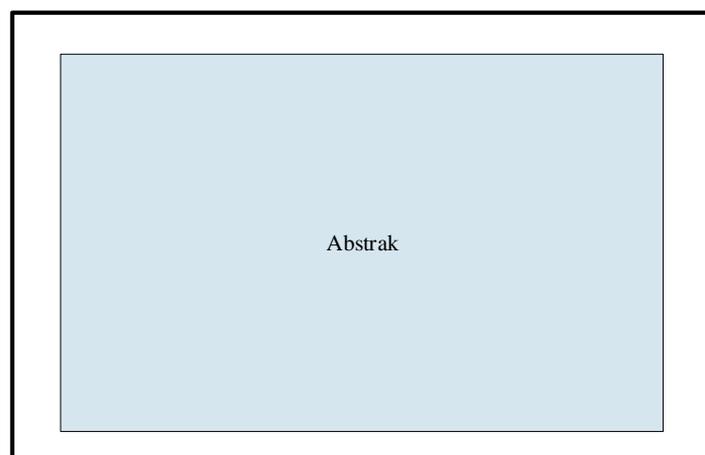
Tampilan menu LSB memiliki beberapa bagian antara lain:

- Label-label
- Textbox pesan yang disisip dan diekstrak
- Textbox daftar karakter yang akan diubah menjadi biner
- Tombol-tombol
- Log

Pada menu LSB, pengguna membuka file audio untuk pertama sekali untuk tempat penyimpanan pesan. Pengguna kemudian memasukkan pesan, dan pesan tersebut akan disisipkan pada file tersebut dengan menekan tombol Sisip.

3.3.3. Menu Abstrak

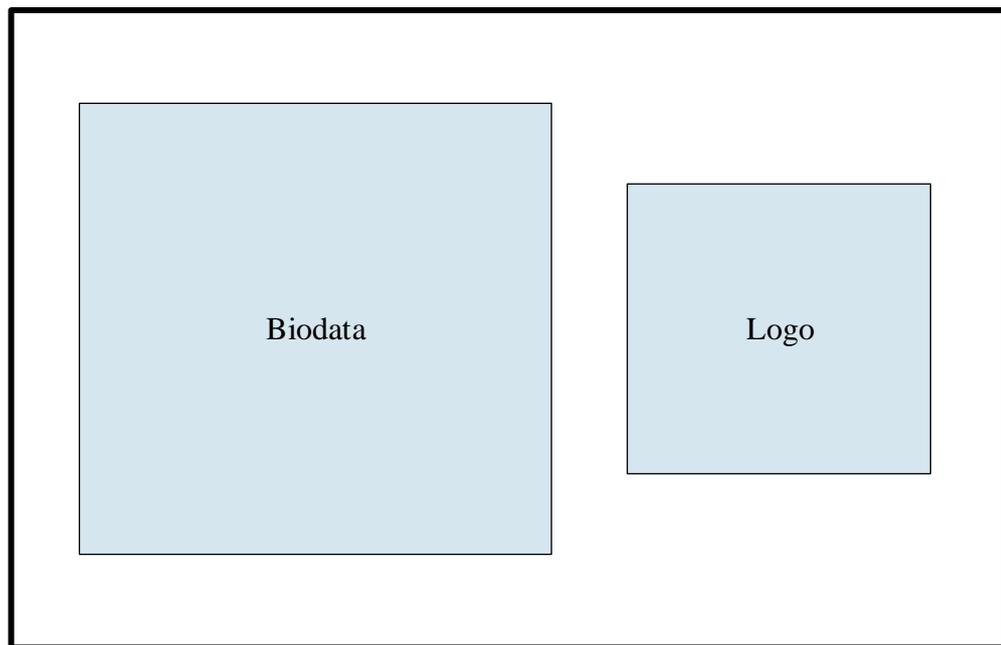
Menu ini menampilkan abstrak tentang penelitian yang penulis lakukan, informasi tentang algoritma LSB dalam menyisipkan pesan dalam file audio. Gambar berikut ini adalah hasil perancangan menu Info.



Gambar 3.8 Tampilan Abstrak

3.3.4. Menu About

Menu ini menampilkan sekilas info tentang penulis. Dalam form ini ada dua buah objek yaitu biodata dan logo. Logo menerangkan institusi dimana penulis melakukan penelitian. Gambar 3.9 adalah hasil tampilan dari menu About.



Gambar 3.9 Tampilan Menu About

BAB IV

HASIL DAN PEMBAHASAN

Hasil dan pembahasan adalah pelaksanaan dan pengujian kebenaran dari perancangan penelitian. Bagian ini merupakan praktek dari program aplikasi yang telah dibuat. Implementasi merupakan tindakan pembuktian yang harus mengikuti pemikiran awal agar suatu sistem yang sudah dirancang benar-benar terjadi. Untuk menjamin agar perancangan berjalan dengan baik maka perlu dilakukan pengujian terhadap sistem dengan menggunakan program aplikasi yang sudah dibuat dengan menggunakan Microsoft Visual Basic.Net. Beberapa komponen utama diperlukan dalam membantu kelancaran program aplikasi tersebut. Komponen-komponen ini mencakup perangkat keras (*hardware*), perangkat lunak (*software*).

4.1 Spesifikasi Sistem

Spesifikasi sistem menjelaskan persyaratan operasional dan kinerja suatu sistem, seperti komputer. Ini dianggap sebagai dokumen tingkat tinggi yang menentukan fungsi global. Spesifikasi sistem membantu untuk menentukan pedoman operasional dan kinerja untuk suatu program aplikasi. Spesifikasi sistem dapat menguraikan bagaimana sistem diharapkan untuk melakukan, dan apa yang mungkin termasuk. Spesifikasi utama dapat mencakup definisi antarmuka, aturan desain dokumen, dan area fungsional. Spesifikasi dapat menentukan akses keamanan. Spesifikasi pada penelitian ini terdiri dari perangkat keras dan lunak.

4.1.1 Spesifikasi Perangkat Keras

Penyisipan pesan dengan algoritma LSB dengan menggunakan media audio sebagai perantara membutuhkan perangkat keras untuk menjalankan sistem. Hal ini sebagai sarana pendukung yang sangat vital. Tabel 4.1 adalah spesifikasi perangkat keras yang digunakan pada penelitian ini.

Tabel 4.1 Spesifikasi perangkat keras

| No. | Komponen | Spesifikasi |
|-----|-----------|-----------------------|
| 1 | Processor | Intel Core i5 2.4 GHz |
| 2 | RAM | 8192 MB |
| 3 | Storage | 500 GB |
| 4 | Display | 14 inch |

4.1.2 Spesifikasi Perangkat Lunak

Tahap spesifikasi perangkat lunak memiliki tujuan, deskripsi kebutuhan dan persiapan agar hasil dan pembahasan dapat terlaksana. Perangkat lunak diperlukan untuk melakukan pemrograman dan penyusunan penelitian agar dapat berjalan dengan semestinya. Kebutuhan akan perangkat lunak sebagai sarana kedua dalam menentukan hasil penelitian. Tabel 4.2 adalah spesifikasi perangkat lunak yang digunakan pada penelitian ini.

Tabel 4.2 Spesifikasi perangkat lunak

| No. | Komponen | Spesifikasi |
|-----|-----------------|---------------------------------|
| 1 | Sistem Operasi | Windows 10 64 Bit |
| 2 | IDE Pemrograman | Microsoft Visual Basic.NET 2010 |
| 3 | Tangkap Gambar | Snipping Tool |
| 4 | Data Editor | Microsoft Excel |

4.2 Implementasi Antarmuka

Implementasi antarmuka adalah suatu tindakan pembuktian dengan menampilkan hasil program aplikasi sehingga pengguna dapat berinteraksi dengan sistem yang sudah dirancang. Penulis memberikan tampilan yang mudah dimengerti agar pengguna tidak mengalami kendala dalam menggunakan program aplikasi steganografi tersebut. Implementasi antarmuka dibagi menjadi empat buah form yang menampilkan submenu dengan fungsi yang berbeda-beda.

4.2.1 Halaman Menu Utama

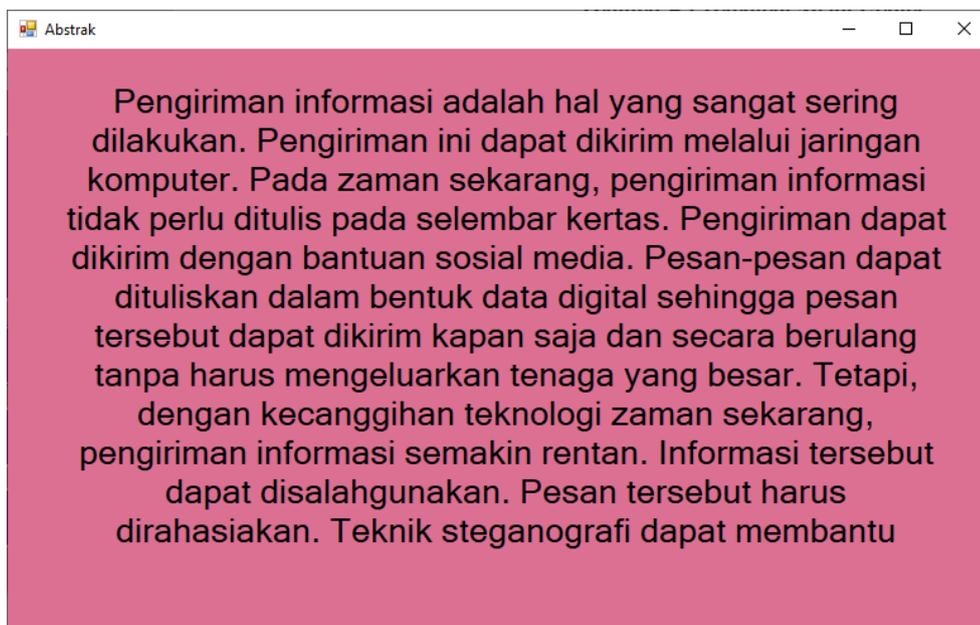
Halaman Menu Utama merupakan halaman utama sebuah program aplikasi dimana setiap pengguna dapat menentukan kemana mereka akan bernavigasi. Pada menu utama, terdapat empat buah tombol yang dapat mengarahkan pengguna menuju submenu yang berbeda-beda. Tiap submenu diwakili oleh fungsi dan form yang berbeda-beda juga. Gambar 4.1 adalah hasil tampilan menu utama.



Gambar 4.1 Halaman Menu Utama

4.2.2 Halaman Abstrak

Halaman info adalah halaman yang menampilkan abstrak dari penelitian yang dilakukan penulis. Pada abstrak akan dijelaskan secara singkat tentang hasil penelitian. Pada abstrak, akan digambarkan latar belakang penelitian, rumusan tujuan dan manfaat penelitian serta gambaran singkat hasil penelitian yang sudah dicapai dengan menggunakan program aplikasi yang telah dibuat. Gambar 4.2 adalah hasil tampilan dari halaman abstrak.



Gambar 4.2 Halaman Abstrak

4.2.3 Halaman About

Halaman about menampilkan biodata penulis dan universitas dimana penulis melaksanakan skripsi. Halaman ini terdiri dari objek label dan *picturebox*. Gambar 4.3 adalah tampilan dari halaman About.



Gambar 4.3 Halaman About

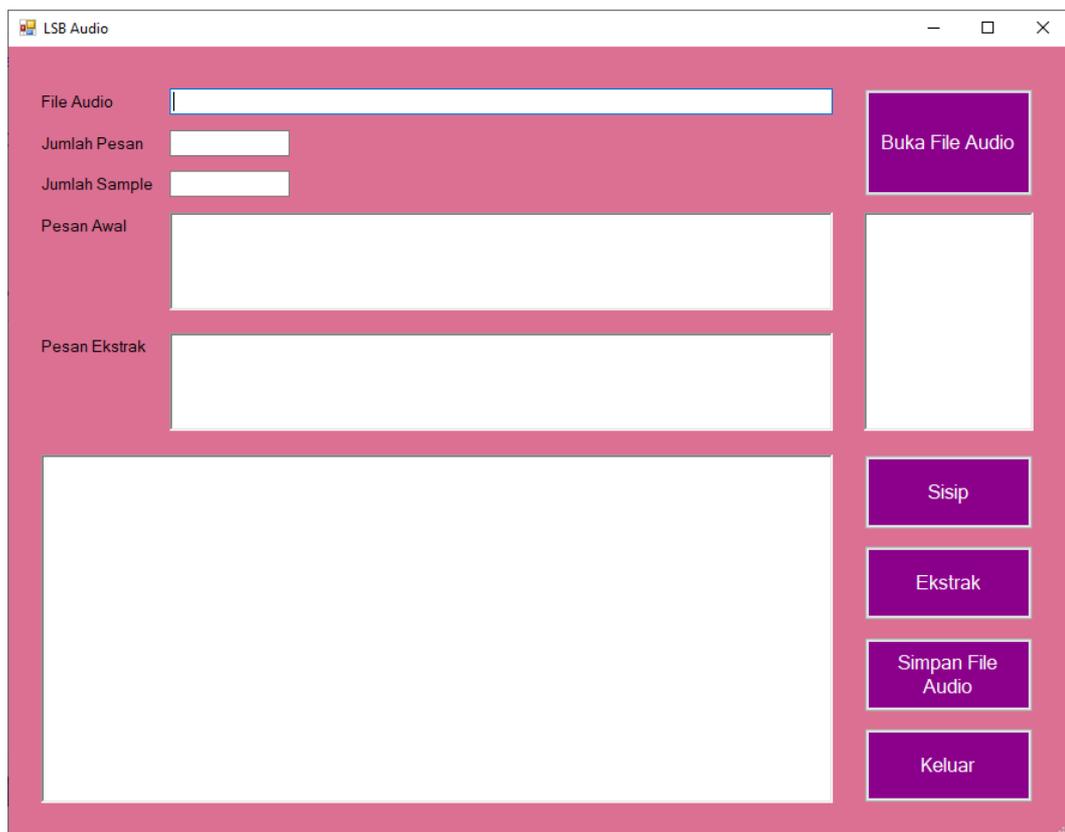
4.2.4 Halaman LSB

Halaman LSB adalah dimana form dari program utama ditampilkan. Halaman ini berfungsi untuk melaksanakan penyisipan pesan pada file dan juga pengambilan data dari file itu kembali sehingga menampilkan pesan yang seperti semula. Pertama-tama, pengguna harus memilih file audio yang akan dijadikan tempat penyimpanan, apakah file .mp3 atau .wav. Pesan yang akan disisipkan akan diketik pada textbox yang sudah disediakan. Setelah pesan diketik, maka pesan tersebut akan disisipkan dengan cara menekan tombol Sisip. Setelah pesan berhasil disisip, maka file dapat disimpan menjadi file stego.

Proses ekstraksi akan bekerja dengan cara yang berlawanan. Pertama pengguna harus membuka file hasil sisip. Setelah file berhasil dimuat dalam program aplikasi, pengguna cukup menekan tombol Ekstrak dan pesan yang

tersimpan pada file tersebut akan ditampilkan pada textbox yang telah ditentukan.

Gambar 4.4 adalah hasil halaman LSB tersebut.

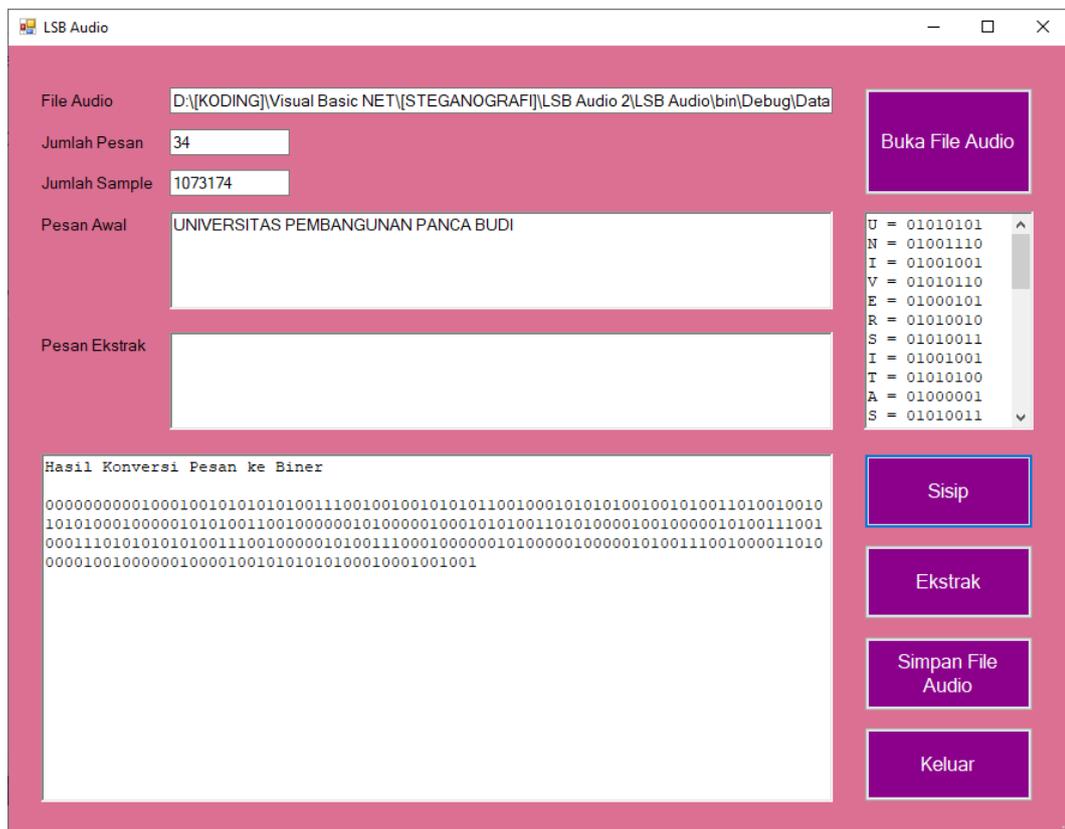


Gambar 4.4 Halaman menu LSB

4.2.5 Hasil Penyisipan

Penyisipan dilakukan dengan menentukan file yang akan digunakan sebagai stego file. Pada saat file dimuat, jumlah sample yang dapat ditampung pada file tersebut akan ditampilkan pada textbox. Pesan akan diketik pada textbox Pesan Awal dan panjang pesan akan tertampil pada textbox Jumlah Pesan. Setelah itu, proses penyisipan dilakukan dengan cara menekan tombol Sisip. Jika program tidak menghasilkan error, maka pesan tersebut dapat disimpan ke dalam file yang

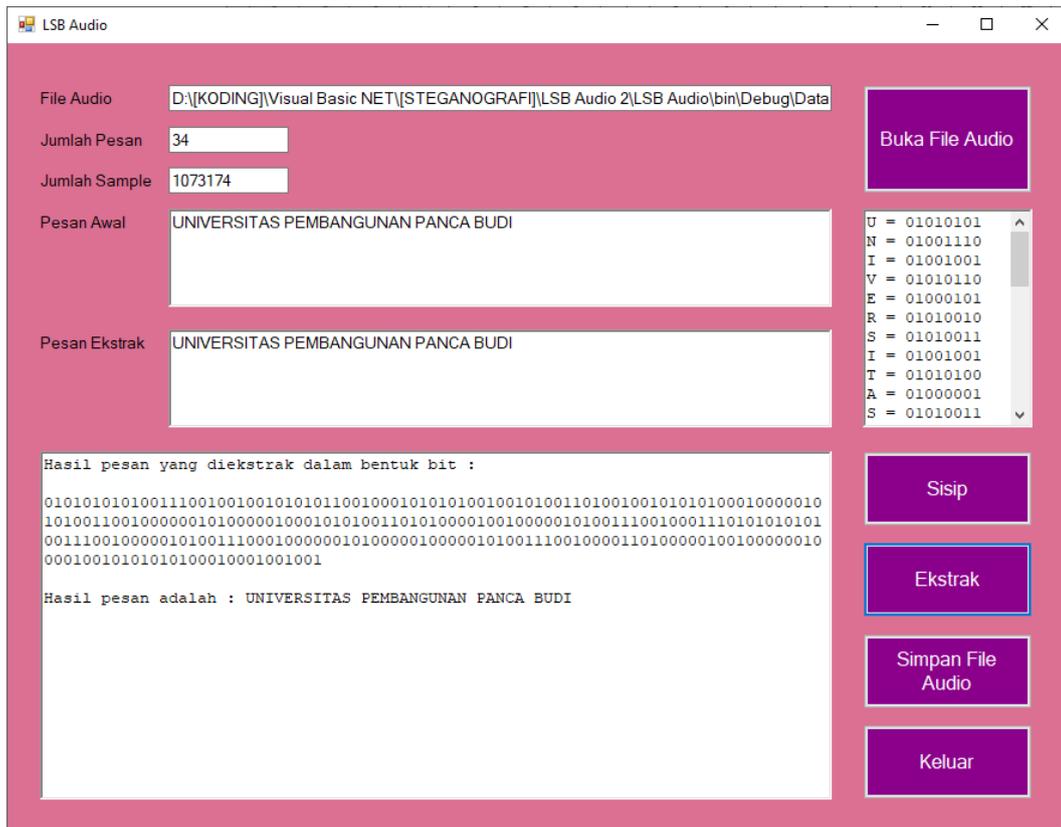
sebelumnya dimuat dengan menekan tombol Simpan. Gambar 4.5 adalah tampilan dari hasil proses penyisipan pesan.



Gambar 4.5 Hasil penyisipan pesan ke dalam file

4.2.6 Hasil Ekstraksi

Proses ekstraksi adalah kebalikan dari proses penyisipan. Pada proses ini, pengguna harus membuka file yang sebelumnya sudah disisip oleh pesan. Setelah terbuka, pengguna hanya tinggal menekan tombol Ekstrak dan jika file tersebut tidak error ada memang benar file yang sudah disisip sebelumnya, maka pesan akan tampil pada textbox Pesan Ekstrak. Gambar 4.6 adalah tampilan dari hasil proses ekstraksi pesan dari file yang sebelumnya sudah disisip oleh pesan.



Gambar 4.6 Hasil ekstraksi pesan dari file

BAB V

PENUTUP

5.1 Kesimpulan

Dalam melakukan penelitian yang berhubungan dengan penyembunyian pesan dalam file audio, penulis dapat menarik beberapa kesimpulan, antara lain:

1. Pesan dapat dikirimkan dengan bantuan media audio sehingga memberikan keamanan dan kenyamanan.
2. File audio dapat berupa .mp3 dan .wav sehingga memberikan keluwesan kepada pengguna dalam memilih file audio sebagai tempat penyimpanan.
3. Algoritma LSB bekerja dengan bit pertama pada sinyal audio sehingga tidak menimbulkan kecurigaan terhadap file yang telah disisip dengan pesan.

5.2 Saran

Setiap penelitian belum tentu sempurna. Ada kekurangan di setiap penelitian yang dilakukan. Penulis memberikan beberapa saran yang dapat dikembangkan di kemudian hari. Saran-saran tersebut antara lain:

1. Sebaiknya penyisipan dapat ditambahkan dengan password agar dapat meningkatkan keamanan.
2. Sebaiknya dapat disisipkan pada media lainnya seperti video atau file.

DAFTAR PUSTAKA

- Aryza, S., Irwanto, M., Lubis, Z., Siahaan, A. P. U., Rahim, R., & Furqan, M. (2018). A Novelty Design Of Minimization Of Electrical Losses In A Vector Controlled Induction Machine Drive. In IOP Conference Series: Materials Science and Engineering (Vol. 300, No. 1, p. 012067). IOP Publishing.
- Barus, S., Sitorus, V. M., Napitupulu, D., Mesran, M., & Supiyandi, S. (2018). Sistem Pendukung Keputusan Pengangkatan Guru Tetap Menerapkan Metode Weight Aggregated Sum Product Assesment (WASPAS). *Jurnal Media Informatika Budidarma*, 2(2).
- Barone, L., Williams, J., & Micklos, D. (2017). Unmet needs for analyzing biological big data: A survey of 704 NSF principal investigators. *PLOS Computational Biology*, 13(10), e1005755. <https://doi.org/10.1371/journal.pcbi.1005755>
- Cokro, A. (2016). Belajar Kriptografi dan Steganografi.
- Fachri, B. (2018). Perancangan Sistem Informasi Iklan Produk Halal Mui Berbasis Mobile Web Menggunakan Multimedia Interaktif. *Jurasik (Jurnal Riset Sistem Informasi dan Teknik Informatika)*, 3, 98-102.
- Fachri, B. (2018, September). APLIKASI PERBAIKAN CITRA EFEK NOISE SALT & PAPPER MENGGUNAKAN METODE CONTRAHARMONIC MEAN FILTER. In *Seminar Nasional Royal (SENAR)* (Vol. 1, No. 1, pp. 87-92).
- 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.
- Girsang, A. S. (2017). Steganografi Dengan Least Significant Bit (LSB). Retrieved from Binus University website: <https://mti.binus.ac.id/2017/10/11/steganografi-dengan-least-significant-bit-lsb-2/>
- Gurevich, Y. (2012). *What Is an Algorithm?* https://doi.org/10.1007/978-3-642-27660-6_3
- Hendini., A. (2016). Pemodelan UML Sistem Informasi Monitoring Penjualan Dan Stok Barang. *Jurnal Khatulistiwa Informatika*, 4(2), 107–116. <https://doi.org/10.31294/jki.v4i2.1262.g1027>
- Khairul, K., IlhamiArsyah, U., Wijaya, R. F., & Utomo, R. B. (2018, September). IMPLEMENTASI AUGMENTED REALITY SEBAGAI MEDIA PROMOSI PENJUALAN RUMAH. In *Seminar Nasional Royal (SENAR)* (Vol. 1, No. 1, pp. 429-434).

- Kurniawan, H. (2018). Pengenalan Struktur Baru untuk Web Mining dan Personalisasi Halaman Web. *Jurnal Teknik dan Informatika*, 5(2), 13-19.
- Kurniawan, T. A. (2018). Pemodelan Use Case (UML): Evaluasi Terhadap beberapa Kesalahan dalam Praktik. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 5(1), 77. <https://doi.org/10.25126/jtiik.201851610>
- Ladjamudin, A.-B. bin. (2005). *Analisis dan Desain Sistem Informasi*. Yogyakarta: Graha Ilmu.
- Nakatsu, R. T. (2009). *Reasoning with Diagrams: Decision-Making and Problem- Solving with Diagrams*. John Wiley & Sons.
- Pratiwi. (2017). Sejarah Steganografi. Retrieved December 22, 2019, from Perbanas Institute website: <https://dosen.perbanas.id/sejarah-steganografi/>
- Putera, A., Siahaan, U., & Rahim, R. (2016). Dynamic key matrix of hill cipher using genetic algorithm. *Int. J. Secur. Its Appl*, 10(8), 173-180.
- Putra, Randi Rian, and Cendra Wadisman. "Implementasi Data Mining Pemilihan Pelanggan Potensial Menggunakan Algoritma K Means." *INTECOMS: Journal of Information Technology and Computer Science* 1.1 (2018): 72-77.
- Rahim, R., Supiyandi, S., Siahaan, A. P. U., Listyorini, T., Utomo, A. P., Triyanto, W. A., ... & Khairunnisa, K. (2018, June). TOPSIS Method Application for Decision Support System in Internal Control for Selecting Best Employees. In *Journal of Physics: Conference Series* (Vol. 1028, No. 1, p. 012052). IOP Publishing.
- Rao, R. V., & Selvamani, K. (2015). Data Security Challenges and Its Solutions in Cloud Computing. *Procedia Computer Science*, 48, 204–209. <https://doi.org/10.1016/j.procs.2015.04.171>
- Sari, R. D., Supiyandi, A. P. U., Siahaan, M. M., & Ginting, R. B. (2017). A Review of IP and MAC Address Filtering in Wireless Network Security. *Int. J. Sci. Res. Sci. Technol*, 3(6), 470-473.
- Segi3Hijau. (2012). Teknik Steganografi dengan Metode LSB. Retrieved from <https://segi3hijau.wordpress.com/2012/11/01/teknik-steganografi-dengan-metode-lsb/>
- Siahaan, A. P. U., Aryza, S., Nasution, M. D. T. P., Napitupulu, D., Wijaya, R. F., & Arisandi, D. (2018). Effect of matrix size in affecting noise reduction level of filtering.
- Siahaan, MD Lesmana, Melva Sari Panjaitan, and Andysah Putera Utama Siahaan. "MikroTik bandwidth management to gain the users prosperity prevalent." *Int. J. Eng. Trends Technol* 42.5 (2016): 218-222.

- Sidik, A. P. (2018). Algoritma RSA dan Elgamal sebagai Algoritma Tambahan untuk Mengatasi Kelemahan Algoritma One Time Pad pada Skema Three Pass Protocol.
- Sun, Y., Zhang, J., Xiong, Y., & Zhu, G. (2014). Data Security and Privacy in Cloud Computing. *International Journal of Distributed Sensor Networks*, 10(7), 190903. <https://doi.org/10.1155/2014/190903>
- Tasril, V. (2018). Sistem Pendukung Keputusan Pemilihan Penerimaan Beasiswa Berprestasi Menggunakan Metode Elimination Et Choix Traduisant La Realite. *INTECOMS: Journal of Information Technology and Computer Science*, 1(1), 100-109.
- Tasril, V., Wijaya, R. F., & Widya, R. (2019). APLIKASI PINTAR BELAJAR BIMBINGAN DAN KONSELING UNTUK SISWA SMA BERBASIS MACROMEDIA FLASH. *Jurnal Informasi Komputer Logika*, 1(3).
- Wasserkrug, S., Dalvi, N., Munson, E. V., Gogolla, M., Sirangelo, C., Fischer - Hübner, S., ... Snodgrass, R. T. (2009). Unified Modeling Language. In *Encyclopedia of Database Systems* (pp. 3232–3239). https://doi.org/10.1007/978-0-387-39940-9_440
- Wibowo, H. R. (2014). *Visual Basic Database*. Yogyakarta: Jubilee Enterprise.
- Zhang, D., Tsotras, V. J., Levialdi, S., Grinstein, G., Berry, D. A., Gouet-Brunet, V., ... Pitoura, E. (2009). Indexed Sequential Access Method. In *Encyclopedia of Database Systems* (pp. 1435–1438). https://doi.org/10.1007/978-0-387-39940-9_738