PENINGKATAN KEAMANAN PESAN TEXT MENGGUNAKAN METODE XOR DENGAN ALGORITMA VERNAM

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	: TAUFIQURRAHMAN CANIAGO
NPM	: 1414370350
PROGRAM STUDI	: SISTEM KOMPUTER

FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS PEMBANGUNAN PANCA BUDI
MEDAN
2019

LEMBAR PENGESAHAN

PENINGKATAN KEAMANAN PESAN TEXT MENGGUNAKAN METODE XOR DENGAN ALGORITMA VERNAM

Disusun Oleh :

NAMA	: TAUFIQURRAHMAN CANIAGO
N P M	: 1414370359
PROGRAM STUDI	: SISTEM KOMPUTER

Skripsi telah disetujui oleh Dosen Pembimbing Skripsi
Pada tanggal 23 Agustus 2019 :

Dosen Pembimbing I	Dosen Pembimbing II
Andysah Putera Utama S, S.Kom., M.Kom., Ph.d	Sri Wahyuni, S.Kom., M.Kom

Mengetahui,

Dekan Sains & Teknologi	Ketua Program Studi
Sri Shindi Indira, S.T., M.Sc	Dr. Muhammad Iqbal, S.Kom., M.Kom

SURAT PERNYATAAN

Saya yang bertanda tangan di bawah ini :

Nama : TAUFIQURRAHMAN CANIAGO
NPM : 1414370350
Prodi : SISTEM KOMPUTER
Konsentrasi : KEAMANAN JARINGAN KOMPUTER
Judul Skripsi : PENINGKATAN KEAMANAN PESAN TEXT MENGGUNAKAN METODE XOR DENGAN ALGORITMA VERNAM

Dengan ini menyatakan bahwa :

1. Tugas Akhir/Skripsi saya bukan hasil Plagiat
2. Sayat tidak akan menuntut perbaikan nilai indeks Prestasi Kumulatif (IPK) setelah ujian Sidang Meja Hijau
3. Skripsi saya dapat dipublikasikan oleh pihak lembaga, dan saya tidak akan menuntut akibat publikasi tersebut

Demikian pernyataan ini saya perbuat dengan sebenar-benarnya, terima kasih

Medan,

Yang membuat pernyataan

TAUFIQURRAHMAN CANIAGO

UNIVERSITAS PEMBANGUNAN PANCA BUDI

FAKULTAS SAINS & TEKNOLOGI

Jl. Jend. Gatot Subroto Km. 4,5 Telp (061) 8455571
website : www.pancabudi.ac.id email: unpab@pancabudi.ac.id
Medan - Indonesia

versitas : Universitas Pembangunan Panca Budi
ultas : SAINS & TEKNOLOGI
sen Pembimbing I : Analysah Putra Utama Siahaan, S.Kom., M.Kom, Ph.D.
sen Pembimbing II : Sri Wahyuni, S.Kom, M.Kom
ma Mahasiswa : TAUFIQURRAHMAN CANIAGO
usan/Program Studi : Sistem Komputer
mor Pokok Mahasiswa : 1414370350
jang Pendidikan : S1
ul Tugas Akhir/Skripsi : Peningkatan Keamanan Pesan Teks Menggunakan Metode XOR Dengan Algoritma VERNAM

TANGGAL	PEMBAHASAN MATERI	PARAF	KETERANGAN
/9 2018	Rev Bab I		
/9 2017	Revisi Bab II		
/9 2017	Revisi Bab II, III		
/9 2017	Revisi Bab IV		
/9 2017	Revisi Bab IV, V		
5/10	Acc Seminar		
2/10	Acc Sidang		
2/7 2017	Acc Jilid		

Medan, 09 September 2018
Diketahui/Disetujui oleh :
Dekan,

Sri Shindi Indira, S.T.,M.Sc.

yang tidak perlu

UNIVERSITAS PEMBANGUNAN PANCA BUDI
FAKULTAS SAINS & TEKNOLOGI
Jl. Jend Gatot Subroto Km. 4,5 Telp (061) 8455571
website : www.pancabudi.ac.id email: unpab@pancabudi.ac.id
Medan - Indonesia

versitas : Universitas Pembangunan Panca Budi
ultas : SAINS & TEKNOLOGI
sen Pembimbing I : Andysah Putra Utama Siahaan, S.Kom., M.Kom., Ph.D.
sen Pembimbing II : Sri Wahyuni., S.Kom., M.Kom
ma Mahasiswa : TAUFIQURRAHMAN CANIAGO
san/Program Studi : Sistem Komputer
mor Pokok Mahasiswa : 1414370350
jang Pendidikan : S1
ul Tugas Akhir/Skripsi : Peningkatan Keamanan Pesan Text Menggunakan Metode XOR Dengan Algoritma VERNAM

TANGGAL	PEMBAHASAN MATERI	PARAF	KETERANGAN
/9 - 2018	Acc judul		
9 - 2018	Revisi Bab I, lgt Bab II		
9 - 2018	Acc Bab I, Revisi Bab II, lgt Bab III		
9 - 2018	Revisi Bab II, Revisi Bab III, lgt Bab IV		
9 - 2018	Acc Bab II, Revisi Bab III, Bab IV		
	lgt Bab V, daftar pustaka		
5 - 2018	Acc Seminar		
0 - 2018	Acc sidang		
7 - 2019	Acc jilid		

Medan, 03 September 2018
Diketahui/Disetujui oleh :
Dekan,

Sri Shinci Indira, S.T., M.Sc.

ang tidak perlu

SURAT BEBAS PUSTAKA

No. 1652 /PERP/BP/2019

Dinyatakan tidak ada sangkut paut dengan UPT. Perpustakaan

Medan, 25 JAN 2019

Ka. UPT. Perpustakaan

Dody Arisandi HP, A.Md

Hal : Permohonan Meja Hijau

FM-BPAA-2012-041

Medan, 22 Januari 2019
Kepada Yth : Bapak/Ibu Dekan
Fakultas SAINS & TEKNOLOGI
UNPAB Medan
Di -
Tempat

Telah Diperiksa oleh LPMU dengan Plagiarisme 49 %
Medan, 30 JANUARI 2019
AN Ka. LPMU
HUSNI M. RITONGA, BA, MSC.

Dengan hormat, saya yang bertanda tangan di bawah ini :

Nama	: TAUFIQURRAHMAN CANIAGO
Tempat/Tgl. Lahir	: Gunungsitoli / 12 Juni 1995
Nama Orang Tua	: ZULKIFLI CANIAGO
N. P. M	: 1414370350
Fakultas	: SAINS & TEKNOLOGI
Program Studi	: Sistem Komputer
No. HP	: 082165819662
Alamat	: Jl.GATOT SUBROTO

Datang bermohon kepada Bapak/Ibu untuk dapat diterima mengikuti Ujian Meja Hijau dengan judul **PENINGKATAN Keamanan PESAN TEXT MENGGUNAKAN Metode XOR Dengan ALGORITMA VERNAM**, Selanjutnya saya menyatakan :

1. Melampirkan KKM yang telah disahkan oleh Ka. Prodi dan Dekan
2. Tidak akan menuntut ujian perbaikan nilai mata kuliah untuk perbaikan indek prestasi (IP), dan mohon diterbitkan ijazahnya setelah lulus ujian meja hijau.
3. Telah tercap keterangan bebas pustaka
4. Terlampir surat keterangan bebas laboratorium
5. Terlampir pas photo untuk ijazah ukuran 4x6 = 5 lembar dan 3x4 = 5 lembar Hitam Putih
6. Terlampir foto copy STTB SLTA dilegalisir 1 (satu) lembar dan bagi mahasiswa yang lanjutan D3 ke S1 lampirkan ijazah dan transkipnya sebanyak 1 lembar.
7. Terlampir pelunasan kwintasi pembayaran uang kuliah berjalan dan wisuda sebanyak 1 lembar
8. Skripsi sudah dijilid tux 2 examplar (1 untuk perpustakaan, 1 untuk mahasiswa) dan jilid kertas jeruk 5 examplar untuk penguji (bentuk dan warna penjilidan diserahkan berdasarkan ketentuan fakultas yang berlaku) dan lembar persetujuan sudah di tandatangani dosen pembimbing, prodi dan dekan
9. Soft Copy Skripsi disimpan di CD sebanyak 2 disc (Sesuai dengan Judul Skripsinya)
10. Terlampir surat keterangan BKKOL (pada saat pengambilan ijazah)
11. Setelah menyelesaikan persyaratan point-point diatas berkas di masukan kedalam MAP
12. Bersedia melunaskan biaya-biaya uang dibebankan untuk memproses pelaksanaan ujian dimaksud, dengan perincian sbb :

1.	[102] Ujian Meja Hijau	: Rp.	100.000
2.	[170] Administrasi Wisuda	: Rp.	1,500,000
3.	[202] Bebas Pustaka	: Rp.	100,000
4.	[221] Bebas LAB	: Rp.	5,000
	Total Biaya	: Rp.	1,705,000

WR 25/01-19

Diketahui/Disetujui oleh :

Sri Shindi Indira, S.T., M.Sc.
Dekan Fakultas SAINS & TEKNOLOGI

Hormat saya

TAUFIQURRAHMAN CANIAGO
1414370350

Catatan :

- 1.Surat permohonan ini sah dan berlaku bila :
  - a. Telah dicap Bukti Pelunasan dari UPT Perpustakaan UNPAB Medan.
  - b. Melampirkan Bukti Pembayaran Uang Kuliah aktif semester berjalan
- 2.Dibuat Rangkap 3 (tiga), untuk - Fakultas - untuk BPAA (asli) - Mhs.ybs.

Plagiarism Detector v. 1092 - Originality Report:

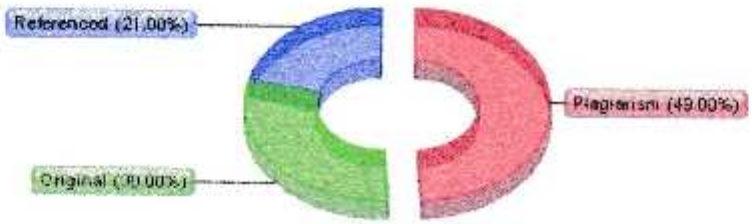
Analyzed document: 30-01-19 2:55:53 PM

"TAUFIQURRAHMAN CANIAGO_1414370350_SISTEM KOMPUTER.docx"

Licensed to: **Universitas Pembangunan Panca Budi_License2**

UNIVERSITAS PEMBANGUNAN PANCA BUDI

Relation chart:



Distribution graph:

Comparison Preset: Rewrite. Detected language: Indonesian

Top sources of plagiarism:

% 15	wrds: 937	http://informatika.stei.itb.ac.id/~rinaldi.munir/Buku/Kriptografi/Bab-1_Pengantar%20Kripto...
% 11	wrds: 691	http://norrianto-arif.blogspot.com/2012/05/
% 11	wrds: 691	http://norrianto-arif.blogspot.com/2012/05/kriptografi.html

Show other Sources:]

Processed resources details:

215 - Ok / 38 - Failed

Show other Sources:]

Important notes:

Wikipedia:	Google Books:	Ghostwriting services:	Anti-cheating:
[not detected]	[not detected]	[not detected]	[not detected]

Excluded Urls:

UNIVERSITAS PEMBANGUNAN PANCA BUDI
FAKULTAS SAINS & TEKNOLOGI

Jl. Jend. Gatot Subroto Km 4,5 Medan Fax. 061-8458077 PO.BOX : 1099 MEDAN

PROGRAM STUDI TEKNIK ELEKTRO	(TERAKREDITASI)
PROGRAM STUDI TEKNIK ARSITEKTUR	(TERAKREDITASI)
PROGRAM STUDI SISTEM KOMPUTER	(TERAKREDITASI)
PROGRAM STUDI TEKNIK KOMPUTER	(TERAKREDITASI)
PROGRAM STUDI AGROTEKNOLOGI	(TERAKREDITASI)
PROGRAM STUDI PETERNAKAN	(TERAKREDITASI)

PERMOHONAN MENGAJUKAN JUDUL SKRIPSI

Saya yang bertanda tangan di bawah ini :

Nama Lengkap	: TAUFIQURRAHMAN CANIAGO
Tempat/Tgl. Lahir	: gunung sitoli / 12 Juni 1995
Nomor Pokok Mahasiswa	: 1414370350
Program Studi	: Sistem Komputer
Konsentrasi	: Keamanan Jaringan Komputer
Jumlah Kredit yang telah dicapai	: 135 SKS, IPK 2.82

Dengan ini mengajukan judul skripsi sesuai dengan bidang ilmu, dengan judul:

No.	Judul SKRIPSI	Persetujuan
1.	Enkripsi dan Deskripsi Pesan menggunakan citra digital pada Pesan Text Menggunakan Metode Steganografi LSB	☐
2.	Implementasi Metode Operator XOR dalam Penyandian Pesan Text	☐
3.	PENINGKATAN Keamanan PESAN TEXT MENGGUNAKAN Metode XOR Dengan ALGORITMA VERNAM	☑

NB : Judul yang disetujui oleh Kepala Program Studi diberikan tanda ☑

Rektor I,

(Ir. Bhakti Alamsyah, M.T., Ph.D.)

Medan, 29 Agustus 2018
Pemohon,

(Taufiqurrahman Caniago)

Nomor : .....
Tanggal : .....
Disahkan oleh :
Dekan

(Sri Shindi Indira, S.T., M.Sc.)

Tanggal : .....
Disetujui oleh :
Dosen Pembimbing I :

(.....)

Tanggal : .....
Disetujui oleh:
Ka. Prodi Sistem Komputer

(MUHAMMAD IQBAL, S.Kom., M.Kom.)

Tanggal : .....
Disetujui oleh:
Dosen Pembimbing II:

(Sri Wahyuni)

YAYASAN PROF. DR. H. KADIRUN YAHYA
UNIVERSITAS PEMBANGUNAN PANCA BUDI
LABORATORIUM KOMPUTER
Jl. Jend. Gatot Subroto Km 4,5 Sei Sikambing Telp. 061-8455571
Medan - 20122

KARTU BEBAS PRAKTIKUM

Yang bertanda tangan dibawah ini Ka. Laboratorium Komputer dengan ini menerangkan bahwa :

Nama	:	TAUFIQURRAHMAN CANIAGO
N.P.M.	:	1414370350
Tingkat/Semester	:	Akhir
Fakultas	:	SAINS & TEKNOLOGI
Jurusan/Prodi	:	Sistem Komputer

Benar dan telah menyelesaikan urusan administrasi di Laboratoriam Komputer Universitas Pembangunan Panca Budi Medan.

Medan, 25 Januari 2019
Ka. Laboratorium

Fachrul Wadly, S. Kom

ABSTRAK

Kriptografi sebagai salah satu cabang ilmu yang dapat digunakan untuk mengamankan data hingga saat ini terus dikembangkan melalui berbagai algoritma. Beberapa penelitian terkait mengenai kriptografi masih mengguankan media berupa teks saja, image saja, maupun file tertentu saja. Pada penelitian ini akan digunakan media berupa seluruh jenis file sebagai media inputan. Adapun algoritma yang dignuakan yaitu Vernam cipher dan Bit shiffting. Kedua algoritma ini dikenal cepat, mudah dan aman untuk digunakan. Percobaan yang dilakukan menggunakan file notepad serta telah diuji melalui aplikasi yang dibangun dengan Visual Studio 2010 telah menghasilkan proses enkripsi dan dekripsi data yang berjalan dengan baik. File hasil enkripsi dapat dibuka dengan kunci yang telah ditetapkan dan tidak mengalami kerusakan, dan sebaliknya untuk proses dekripsi data juga demikian

Kata Kunci: Kriptografi, Vernam Cipher, File

KATA PENGANTAR

Puji Syukur penulis panjatkan kepada Tuhan Yang Maha Esa, yang telah memberikan rahmat-Nya kepada peneliti, sehingga Skripsi ini dapat diselesaikan oleh peneliti tepat pada waktunya dengan judul Penerapan Kriptografi Sebagai Alternatif Pengamanan Pada Aplikasi.

Skripsi ini dilakukan guna memenuhi salah satu syarat pemenuhan kurikulum dalam menyelesaikan pendidikan pada Program Studi S1 Sistem Komputer Fakultas Sains dan Teknologi Universitas Pembangunan Panca Budi Medan. Pada kesempatan ini, penulis menyampaikan rasa terima kasih dan penghargaan yang sebesar-besarnya kepada :

1. Bapak Dr. H. Muhammad Isa Indrawan, SE, MM, selaku Rektor Universitas Pembangunan Panca Budi Medan.
2. Ibu Sri Shindi Indira, S.T., M.S.c, selaku Dekan Fakultas Sains dan Teknologi Universitas Pembangunan Panca Budi Medan.
3. Bapak Dr. Muhammad Iqbal selaku kaprodi sistem komputer
4. Bapak Andysah Putera Utama Siahaan, S.Kom., M.Kom, .,Ph.D, Selaku Pembimbing I yang juga telah memberikan pengarahan dan petunjuk Skripsi Ini.
5. Ibu Sri Wahyuni S.Kom., M.Kom, selaku Dosen Pembimbing II yang juga telah memberikan pengarahan dan petunjuk dalam Skripsi ini.
6. Bapak/Ibu Dosen beserta seluruh staf Universitas Pembangunan Panca Budi Medan

7. Teristimewa kepada Alm. ayahanda Zulkifli Caniago dan Almh. Ibunda Halwani Piliang, beserta keluarga saya yang telah banyak memberikan bimbingan dan bantuan baik moril maupun material selama penulis mengikuti pendidikan hingga selesainya Skripsi ini.
8. Kepada seluruh rekan di program Studi Sistem Komputer Universitas Pembangunan Panca Budi Medan yang telah memberikan dukungan moril kepada penulis.

Penulis menyadari bahwa Skripsi ini masih kurang sempurna. Oleh karena itu, penulis sangat mengharapkan dan menghargai saran maupun kritikan dari pembaca dan semua pihak yang mengarah kepada perbaikan Skripsi ini.

Medan, Juli 2019
Penulis,

(TAUFIQURRAHMAN CANIAGO)
(NPM. 1414370350)

DAFTAR ISI

Halaman

DAFTAR GAMBAR

No Judul **Hal**

DAFTAR TABEL

No Judul **Hal**

BAB I

PENDAHULUAN

1. Latar Belakang

Kriptografi bagi kebanyakan orang adalah sesuatu yang sangat sulit dan kita sebagai pemula cenderung malas untuk mempelajarinya. Namun ada sebuah metode *kriptografi* yang agak mudah untuk dipelajari dan para ahlipun telah menyatakan bahwa metode ini merupakan metode *kriptografi* yang cukup aman untuk digunakan. Metode tersebut biasa dikenal dengan nama *One Time Pad* (*OTP*) atau yang lebih dikenal dengan sebutan *Vernam Cipher*. *Vernam Cipher* diciptakan oleh *Mayor J. Maugborne* dan *G. Vernam* pada tahun 1917.

Dalam proses *enkripsi*, *algoritma* ini menggunakan cara *stream cipher* yang berasal dari hasil *XOR* antara *bit plaintext* dan *bit key*. Pada metode ini *plaintext* diubah kedalam kode *ASCII* dan kemudian dikenakan operasi *XOR* terhadap kunci yang sudah diubah ke dalam kode *ASCII*.

Algoritma dari *enkripsi* adalah fungsi-fungsi yang digunakan untuk melakukan fungsi *enkripsi* dan *dekripsi*. *Algoritma* yang digunakan menentukan kekuatan dari *enkripsi*, dan ini biasanya dibuktikan dengan basis matematika. Berdasarkan cara memproses teks (*plaintext*), *cipher* dapat dikategorikan menjadi dua jenis: *block cipher* and *stream cipher*. *Block cipher* bekerja dengan memproses data secara blok, dimana beberapa karakter / data digabungkan menjadi satu blok. Setiap proses satu blok menghasilkan keluaran satu blok juga.

Sementara itu *stream cipher* bekerja memproses masukan (karakter atau data) secara terus menerus dan menghasilkan data pada saat yang bersamaan.

Suatu *one-time pad* diciptakan dengan men-*generate* suatu *string* yang terdiri dari karakter-karakter atau angka-angka yang panjangnya harus *minimal* sama dengan kata terpanjang dalam pesan yang akan di*enkripsi*kan. *String* ini di-*generate* secara acak atau *random*, misalnya dengan menggunakan *random number generator* pada *computer*. *String* tersebut kemudian dituliskan pada suatu *pad*. *Pad-pad* tersebut kemudian diberikan kepada siapapun yang ingin menggunakannya untuk mengirim ataupun menerima pesan.

Aplikasi yang akan dibuat oleh penulis adalah dengan menggunakan *visual studio 2010* dengan menggunakan *kriptografi one time pad* agar dapat meng*enkripsi* dan *dekripsi* data teks yang akan digunakan secara rahasia dan lebih mudah digunakan oleh *user* nantinya. Berdasarkan latar belakang di atas maka penulis tertarik untuk memilih judul "**Peningkatan Keamanan Pesan *Text* Menggunakan Metode *XOR* Dengan *Algoritma Vernam*".**

2. Rumusan Masalah

Berdasarkan latar belakang masalah di atas maka rumusan masalah adalah sebagai berikut :

a. Bagaimana merancang sebuah *software* pengamanan informasi teks dengan menggunakan *Visual Studio .NET?*
b. Bagaimana membuat *enkripsi* dan *dekripsi* informasi dengan menggunkanan *algoritma vernam chiper* dengan metode *XOR* ?

3. Batasan Masalah

Dalam perancangan *aplikasi kriptografi* ini penulis membatasi masalah sebagai berikut :

a. Metode yang digunakan pada perancangan *aplikasi* pengamanan informasi ini menggunakan metode *XOR* untuk *ekripsi* dan *dekripsi text*.
b. Bahasa program yang digunakan dalam perancangan *aplikasi kriptografi one time pad* ini adalah *Visual Studio .NET*.
c. Dalam proses *enkripsi* dan *deskripsi* hanya menggunakan *file* berformat *.txt* (*notepad*).

4. Tujuan Penelitian

Tujuan yang ingin dicapai penulis dalam perancangan *aplikasi kriptografi* ini adalah :

a. Untuk mengubah pengiriman data.
b. Membuat suatu *aplikasi kriptografi* yang meng*implementasi*kan *algoritma XOR* dan *Algoritma Vernam* sehingga dapat mengatasi masalah keamanan informasi serta menjaga kerahasiaan data.

5. Manfaat Penelitian

Perancangan *aplikasi kriptografi* ini bermanfaat bagi masyarakat luas antara lain :

a. Mempermudah bagi pengguna untuk meng*enskripsi* data teks yang akan digunakan secara rahasia.

b. *Aplikasi kriptografi* ini dapat digunakan oleh semua kalangan masyarakat luas agar dapat membuat teks yang dapat dikunci dengan kata khusus.

6. Metodologi Penelitian

a. Metode Pengumpulan Data

Metode Pengumpulan Data yang digunakan dalam penelitian ini adalah metode *deskriptif*. Adapun teknik pengumpulan data dilakukan dengan cara sebagai berikut:

1) Studi *literature*

   Pengumpulan data dengan cara mengumpulkan *literature*, jurnal, *paper* dan bacaan-bacaan yang ada kaitannya dengan judul penelitian.

2) Studi Pustaka

   Pengumpulan data dengan menggunakan atau mengumpulkan sumber-sumber tertulis, dengan cara membaca, mempelajari dan mencatat hal-hal penting yang berhubungan dengan masalah yang sedang dibahas guna memperoleh gambaran secara teoritis.

7. Sistematika Penulisan

Adapun struktur penulisan pada masing-masing bab dalam laporan tugas akhir ini adalah sebagai berikut:

BAB I PENDAHULUAN

Membahas Latar Belakang Masalah *Kriptografi* dan *Algoritma Vernam Chiper*, Rumusan Masalah dari *Algoritma Vernam* dan Penerapan dalam proses *enkripsi* dan *deskirpsi*, Batasan Masalah, Tujuan dan Manfaat Penelitian *Algoritma Vernam Chiper*, *Metodelogi* Penelitian dan *Sistematika* Penulisan.

BAB II LANDASAN TEORI

Memaparkan teori-teori yang didapat dari sumber-sumber yang *relevan* untuk digunakan sebagai panduan dalam penelitian serta penyusunan skripsi.

BAB III PERANCANGAN SISTEM

Menjelaskan tentang gambaran sistem dari *algoritma vernam chiper* serta *deskripsi* dari *algoritma vernam chiper* hasil *analisis* sistem yang akan dijadikan sebagai petunjuk untuk perancangan sistem selanjutnya sesuai dengan *flowchat* yang diusulkan.

BAB IV *IMPLEMENTASI* SISTEM

Bab ini menguraikan langkah-langkah dalam *implementasi* sistem menggunakan *algoritma vernam chiper*, disertai dengan komponen-komponen kebutuhan sistem dalam proses *enkripsi* dan *deskripsi* dari *algoritma vernam chiper*.

BAB V PENUTUP

Mengemukakan kesimpulan yang diambil dari hasil penelitian dan perancangan sistem, serta saran-saran untuk pengembangan

selanjutnya, agar dapat dilakukan perbaikan-perbaikan dimasa yang akan datang.

BAB II

LANDASAN TEORI

1. *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*)

Jika *transformasi*nya dapat dikembalikan, *kriptografi* juga dapat diartikan sebagai proses mengubah kembali data yang ter*enkripsi* menjadi bentuk yang mudah dipahami. Sehingga, *kriptografi* juga dapat diartikan sebagai proses untuk melindungi data dalam arti yang luas. Pengertian *Kriptografi* dalam kamus bahasa Inggris *Oxford* adalah Sebuah teknik rahasia dalam penulisan, dengan karakter khusus, dengan mengguanakan huruf dan karakter di luar bentuk aslinya, atau dengan metode-metode lain yang hanya dapat dipahami oleh pihak-pihak yang memproses kunci, juga semua hal yang ditulis dengan cara seperti ini. Jadi, secara umun *kriptografi* diartikan sebagai seni menulis atau memecahkan *cipher*.

Dalam perkembangannya, *kriptografi* juga digunakan untuk meng*identifikasi* pengiriman pesan dan tanda tangan *digital* dan keaslian pesan dengan sidik jari *digital*. (*Dony Ariyus, 2005*)

Di dalam *kriptografi* kita akan sering menemukan berbagai istilah atau *terminology*. Beberapa istilah yang harus diketahui yaitu :

a) Pesan, *plaintext*, dan *cipherteks*

Pesan (*message*) adalah data atau informasi yang dapat dibaca dan dimengerti maknanya. Nama lain untuk pesan adalah (*plaintext*) atau *teks* jelas (*cleartext*). Agar pesan tidak dapat dimengerti maknanya oleh pihak lain yang tidak berkepentingan, maka pesan perlu disandikan kebentuk lain yang tidak dapat dipahami. Bentuk pesan yang tersandi disebut *cipherteks* (*ciphertext*) atau *kriptogram* (*cryptogram*). *Cipherteks* harus dapat di*transformasi*kan kembali menjadi *plaintext* semula agar dapat diterima dan bisa dibaca.

b) Pengirim dan penerima

Komunikasi data melibatkan pertukaran pesan antara dua *entitas*. Pengirim (*sender*) adalah *entitas* yang mengirim pesan kepada *entitas* lainnya. Penerima (*receiver*) adalah *entitas* yang menerima pesan. Pengirim tentu menginginkan pesan dapat dikirim secara aman, yaitu pengirim yakin bahwa pihak lain tidak dapat membaca isi pesan yang dikirim. Solusinya adalah dengan cara menyandikan pesan menjadi *cipherteks*.

c) *Enkripsi* dan *dekripsi*

Proses menyandikan *plainteks* menjadi *cipherteks* disebut *enkripsi* (*encryption*) atau *enciphering*. Sedangkan proses mengembalikan *cipherteks* menjadi *plainteks* disebut *dekripsi* (*decryption*) atau *deciphering*.

d) *Cipher* dan kunci

Algoritma *kriptogarfi* disebut juga *cipher*, yaitu aturan untuk *enkripsi* dan *dekripsi*, atau fungsi matematika yang digunakan untuk *enkripsi* dan *dekripsi*. Beberapa *cipher* memerlukan *algoritma* yang berbeda untuk *enciphering* dan *deciphering*. *Kriptografi Asimetri* (*Asymetric Cryptography*) . Konsep *matematis* yang mendasari *algoritma kriptografi* adalah *relasi* antara dua buah himpunan yang berisi *elemen–elemen plainteks* dan himpunan yang berisi *cipherteks*. *Enkripsi* dan *dekripsi* merupakan fungsi yang memetakan *elemen-elemen* antara dua himpunan tersebut. Misalkan *P* menyatakan *plainteks* dan *C* menyatakan *cipherteks*, maka fungsi *enkripsi E* memetakan *P* ke *C*. $E(P) = C$ Dan fungsi *dekripsi D* memetakan *C* ke *P* $D(C) = P$.

Karena proses *enkripsi* kemudian *dekripsi* mengembalikan pesan ke pesan semula, maka kesamaan berikut harus benar, $D(E(P)) = P$

Kriptografi mengatasi masalah keamanan data dengan menggunakan kunci, yang dalam hal ini *algoritma* tidak dirahasiakan lagi, tetapi kunci harus tetap di jaga kerahasiaannya. Kunci (*key*) adalah *parameter* yang digunakan untuk *transformasi enciphering* dan *deciphering*. Kunci biasanya berupa *string* atau deretan bilangan. Dengan menggunakan K, maka fungsi *enkripsi* dan *dekripsi* dapat ditulis sebagai :

$E_K(P) = C \quad D_K(C) = P$

Keterangan :

P = *plainteks*

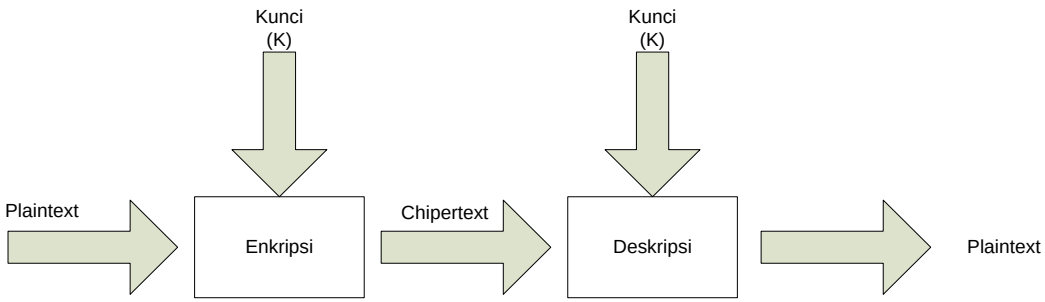
C = *cipherteks*

K = kunci

EK = proses *enkripsi* menggunakan kunci *K*

DK = proses *dekripsi* menggunakan kunci *K*

Skema *enkripsi* dengan menggunakan kunci diperlihatkan pada gambar dibawah ini:



Gambar 1. Skema *enkripsi* dan *dekripsi* dengan menggunakan kunci

Sumber : *Dony Ariyus, 2015*

e) Sistem *kriptografi*

kriptografi membentuk sebuah sistem yang dinamakan sistem *Kriptografi*. *Sistem kriptografi* (*cryptosystem*) adalah kumpulan yang terdiri dari *algoritma kriptografi* semua *plainteks* dan *cipherteks* yang mungkin, dan kunci. Di dalam *kriptografi*, *cipher* hanyalah salah satu *komponen* saja.

f) Penyadap (*eavesdropper*)

adalah orang yang mencoba menangkap pesan selama di*transmisi*kan. Tujuan penyadap adalah untuk mendapatkan informasi sebanyak-banyaknya mengenai sistem *kriptografi* yang digunakan untuk berkomunikasi dengan maksud untuk memecahkan *cipherteks*. Nama lain penyadap : *enemy*, *adversary*, *intruder*, *interceptor*, *bad guy*.

g) *Kriptanalisis* dan *kriptologi*

Kriptografi berkembang sedemikian rupa sehingga melahirkan bidang yang berlawanan yaitu *kriptanalisis*. *Kriptanalisis* (*cryptanalysis*) adalah ilmu dan seni untuk memecahkan *cipherteks* menjadi *plainteks* tanpa mengetahui kunci yang digunakan. Pelakunya disebut *kriptanalis*.

a. 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 men*deteksi 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 di*realisasi*kan 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 meng*identifikasi* kebenaran pihak-pihak yang ber*komunikasi* (*user authentication* atau *entity authentication*) maupun meng*identifikasi* kebenaran sumber pesan (*data origin authentication*). Dua pihak yang saling ber*komunikasi* harus dapat meng*otentikasi* satu sama lain sehingga ia dapat memastikan sumber pesan. Pesan yang dikirim melalui saluran *komunikasi* juga harus di*otentikasi* asalnya. *Otentikasi* sumber pesan secara *implisit* juga memberikan kepastian *integritas* data, sebab jika pesan telah di*modifikasi* berarti sumber pesan sudah tidak benar. Oleh karena itu, layanan *integritas* data selalu di*kombinasi*kan dengan layanan *otentikasi* sumber pesan. Di dalam *kriptografi*, layanan ini di*realisasi*kan dengan menggunakan tanda-tangan *digital* (*digital signature*). Tanda-tangan *digital* menyatakan sumber pesan.

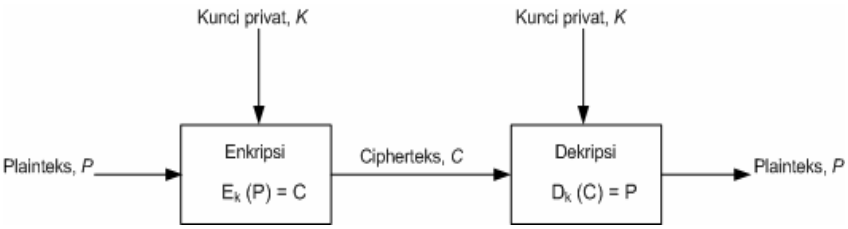
4) *Nirpenyangkalan* (*non-repudiation*) adalah layanan untuk mencegah *entitas* yang berkomunikasi melakukan penyangkalan, yaitu pengirim pesan menyangkal melakukan pengiriman atau penerima pesan menyangkal telah menerima pesan.

b. Jenis-jenis *Kriptografi*

Berdasarkan kunci yang digunakan untuk *enkripsi* dan *dekripsi*, *kriptografi* dapat dibedakan menjadi 2 macam, yaitu *kriptografi simetri* (*symmetric cryptography*) dan *kriptografi asimetri* (*asymetric cryptography).*

1. *Kriptografi Simetri* (*Symetric Cryptography*)

Pada sistem *kriptografi simetri*, kunci untuk proses *enkripsi* sama dengan kunci untuk proses *dekripsi*. Keamanan sistem *kriptografi simetri* terletak pada kerahasiaan kunci. Istilah lain untuk *kriptografi simetri* adalah *kriptografi* kunci *privat* (*private key cryptography*) atau *kriptografi konvensional* (*conventional cryptography*).



Gambar 2 *Kriptografi Simetri* (*Symetric Cryptography*)

Sumber : *Zelvina, 58 : 2012*

Algoritma kriptografi simetri dapat dikelompokkan menjadi dua kategori antara lain :

a. *Cipher* aliran (*stream cipher*)

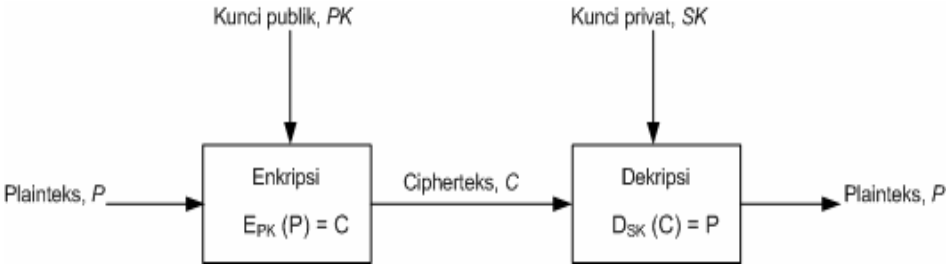
Algoritma kriptografi beroperasi pada *plainteks*/*cipherteks* dalam bentuk *bit* tunggal yang dalam hal ini rangkaian *bit* di*enkripsi*kan/di*dekripsi*kan *bit* per *bit*. *Cipher* aliran meng*enkripsi* satu *bit* setiap kali.

b. *Cipher blok* (*block cipher*)

Algoritma kriptografi beroperasi pada *plainteks* / *cipherteks* dalam bentuk *blok bit*, yang dalam hal ini rangkaian *bit* dibagi menjadi *blok-blok bit* yang panjangnya sudah ditentukan sebelumnya. *Cipher* blok meng*enkripsi* satu blok *bit* setiap kali.

2. *Kriptografi Asimetri* (*Asymetric Cryptography*)

Pada sistem *kriptografi asimetri*, kunci untuk proses *enkripsi* tidak sama dengan kunci untuk proses *dekripsi*. Istilah lain untuk *kriptografi asimetri* adalah *kriptografi* kunci *publik* (*public key cryptography*), sebab kunci untuk *enkripsi* tidak rahasia dan dapat diketahui oleh siapapun, sementara kunci untuk *dekripsi* hanya diketahui oleh penerima pesan.



Gambar 3. *Kriptografi Asimetri* (*Asymetric Cryptography*)

Sumber : *Zelvina, 58 : 2012*

2. Algoritma *Vernam Chiper*

Kriptografi bagi kebanyakan orang adalah sesuatu yang sangat sulit dan kita sebagai pemula cenderung malas untuk mempelajarinya. Namun ada sebuah metode *kriptografi* yang agak mudah untuk dipelajari dan para ahlipun telah

menyatakan bahwa metode ini merupakan metode *kriptografi* yang cukup aman untuk digunakan. Metode tersebut biasa dikenal dengan nama *One Time Pad* (*OTP*) atau yang lebih dikenal dengan sebutan *Vernam Cipher*. *Vernam Cipher* diciptakan oleh *Mayor J. Maugborne* dan *G. Vernam* pada tahun 1917.

Algoritma *One Time Pad* (*OTP*) merupakan *algoritma* berjenis *symetric 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* yang berasal dari hasil *XOR* antara *bit plaintext* dan *bit key*. Pada metode ini *plaintext* diubah kedalam kode *ASCII* dan kemudian dikenakan operasi *XOR* terhadap kunci yang sudah diubah ke dalam kode *ASCII*.

3. Metode *XOR*

Operasi *XOR* merupakan operasi logika *bitwise* yang bekerja dengan membandingkan dua buah *bit* yang apabila pada salah satu *bit* nya bernilai Benar, maka hasil akhir operasi *XOR* tersebut adalah benar. Namun, bila kedua *bit* yang akan dibandingkan bernilai Salah atau keduanya bernilai Benar maka hasil akhir operasi *XOR* tersebut adalah Salah.

XOR enkripsi, meskipun bukan sistem kunci-publik seperti *RSA*, hampir bisa dipecahkan melalui metode *brute force*. Hal ini rentan terhadap pola, tetapi kelemahan ini dapat dihindari melalui, pertama mengompresi *file* (sehingga untuk menghilangkan pola). *Enkripsi eksklusif* atau membutuhkan baik *encryptor* dan *decryptor* memiliki akses ke kunci *enkripsi*, tetapi algoritma *enkripsi*, sementara

sangat sederhana, hampir bisa dipecahkan. Karya *XOR enkripsi* dengan menggunakan fungsi aljabar *boolean XOR*.

Code:

```
X     Y     X^Y
--------------
1     1     0
1     0     1
0     1     1
0     0     0
```

Namun bagaimana jika kita melakukan dua kali operasi *XOR* dua kali terhadap suatu *bit* dengan *operand* yang sama, maka hasilnya akan kembali seperti semua. Seperti contoh gambar berikut.

Code:

```
X     Y     X^Y     (X^Y)^Y
-----------------------
1     1     0       1
1     0     1       1
0     1     1       0
0     0     0       0
```

Dapat dilihat dari kedua gambar di atas, pada gambar pertama terlihat nilai pada *variabel* X yang di *XOR* kan dengan *variabel* Y dan menghasilkan nilai yang ada pada *variabel* X^Y. Namun, jika kita lihat pada gambar kedua, *variabel* X^Y di *XOR* kan lagi dengan *variabel* Y dan kemudian menghasilkan nilai yang sama dengan nilai yang ada pada *variabel* X. Sifat seperti ini yang dapat kita gunakan untuk membuat *enkripsi* sederhana.

4. *One Time Pad*

Algoritma One Time Pad (OTP) merupakan algoritma berjenis *symetric 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* yang berasal dari hasil *XOR* antara *bit plaintext* dan *bit key*. Pada metode ini *plaintext* diubah kedalam kode *ASCII* dan kemudian dikenakan operasi *XOR* terhadap kunci yang sudah diubah ke dalam kode *ASCII*. (*Hamokwarong, 10 :2011*)

One-time pad adalah salah satu *stream cipher* klasik yang secara *matematis* terbukti sempurna aman. *Cipher teks*nya tidak mungkin dapat dipecahkan. Keamanan *algoritma one-time pad* terletak pada penggunaan barisan bilangan acak sejati (*trully random*) sebagai kunci *enkripsi*, panjang kunci sama dengan panjang pesan dan tidak ada perulangan kunci sebagaimana pada pada *Vernam cipher* atau *Vigenere cipher*. (*Munir, 12 :2011*)

Sayangnya *one-time pad* tidak dapat di*implementasi*kan secara *praktis* sebab pembangkitan bilangan acak sejati tidak dapat diulang kembali di sisi penerima pesan. Oleh karena itu kunci (*pad*) harus dikirim melalui saluran komunikasi yang kedua (misalnya melalui kurir), sayangnya saluran kedua itu umumnya lambat dan ongkosnya mahal. *One-time pad* masih dapat diterapkan namun kunci yang berupa barisan bilangan acak diganti dengan barisan bilangan semi-acak (*pseudo-random*) dengan syarat barisan kunci itu tidak boleh berulang. *(Munir, 12 :2011*)

5. Pengertian Informasi

Secara *Etimologi*, kata informasi ini berasal dari kata bahasa Perancis kuno *informacion* (tahun 1387) mengambil istilah dari bahasa Latin yaitu *informationem* yang berarti "konsep, ide atau garis besar". Informasi ini merupakan kata benda dari *informare* yang berarti aktivitas dalam "pengetahuan yang dikomunikasikan".

Informasi adalah hasil pemrosesan data yang diperoleh dari setiap *elemen* sistem menjadi bentuk yang mudah dipahami dan merupakan pengetahuan yang *relevan* dan berguna (*Yulansari, 6 : 2013*).

Informasi bisa menjadi fungsi penting dalam membantu mengurangi rasa cemas pada seseorang. Menurut pendapat *Notoatmodjo* (2008) bahwa semakin banyak memiliki informasi dapat memengaruhi atau menambah pengetahuan terhadap seseorang dan dengan pengetahuan tersebut bisa menimbulkan kesadaran yang akhirnya seseorang itu akan berperilaku sesuai dengan pengetahuan yang dimilikinya.

Informasi adalah data yang telah diolah melalui proses tertentu menjadi sesuatu yang menambah pengetahuan atau temuan yang mempunyai arti baru bagi pemakainya (*Melina, 38 : 2012*).

Adapun fungsi-fungsi informasi adalah sebagai berikut:

a. Untuk meningkatkan pengetahuan bagi si pemakai.
b. Untuk mengurangi ketidakpastian dalam proses pengambilan keputusan pemakai.

c. Menggambarkan keadaan yang sebenarnya dari sesuatu hal. Informasi yang berkualitas harus akurat, tepat dan *relevan*.

Sumber dari informasi adalah data. Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata. Data merupakan bentuk yang masih mentah, belum dapat bercerita banyak sehingga perlu diolah lebih lanjut. Data diolah melalui suatu metode untuk menghasilkan informasi. Data dapat berbentuk simbol-simbol semacam huruf, angka, bentuk suara, sinyal, gambar, dan sebagainya.

6. Definisi *Visual Basic.Net*

Visual Basic .Net merupakan salah satu *tool development Microsoft* yang dapat digunakan untuk membuat *aplikasi* di lingkungan kerja berbasis sistem operasi *Windows*. *Visual Basic .NET* menyediakan *tools* bagi para *developer* untuk membangun *aplikasi* yang berjalan di *.Net Framework* (*safik : 2012 : 2*).

Visual BASIC (*Beginners All-Purpose Symbolic Instruction Code*) merupakan Bahasa pemrograman *Integrated Development Environment* (*IDE*), yaitu bahasa pemrograman *visual* yang digunakan untuk membuat program *aplikasi* atau *software* berbasis sistem operasi *Microsoft Windows*, dengan menggunakan model pemrograman "*Common Object Model* (*COM*)".

Visual basic merupakan turunan bahasa pemrograman *BASIC* yang menawarkan pengembangan perangkat lunak komputer bebasis *grafik* dengan cepat. Dengan menggunakan bahasa pemrograman VB, para *progammer* dapat

membangun *aplikasi* dengan menggunakan *komponen-komponen* yang di sediakan VB.

Microsoft Visual Basic (sering disingkat sebagai VB saja) merupakan sebuah bahasa pemrograman yang menawarkan *Integrated Development Environment* (*IDE*) *visual* untuk membuat program perangkat lunak berbasis sistem operasi *Microsoft Windows* dengan menggunakan model pemrograman (*COM*), *Visual Basic* merupakan turunan bahasa pemrograman *BASIC* dan menawarkan pengembangan perangkat lunak komputer berbasis *grafik* dengan cepat, Beberapa bahasa *skrip* seperti *Visual Basic for Applications* (*VBA*) dan *Visual Basic Scripting Edition* (*VBScript*), mirip seperti halnya *Visual Basic*, tetapi cara kerjanya yang berbeda.

Para *programmer* dapat membangun *aplikasi* dengan menggunakan *komponen-komponen* yang disediakan oleh *Microsoft Visual Basic* Program-program yang ditulis dengan *Visual Basic* juga dapat menggunakan *Windows API*, tapi membutuhkan deklarasi fungsi luar tambahan.

Dalam pemrograman untuk bisnis, *Visual Basic* memiliki pangsa pasar yang sangat luas. Dalam sebuah *survey* yang dilakukan pada tahun 2005, 62% pengembang perangkat lunak dilaporkan menggunakan berbagai bentuk *Visual Basic*, yang diikuti oleh C++, *JavaScript*, C#, dan *Java*.

1) Komponen kerja

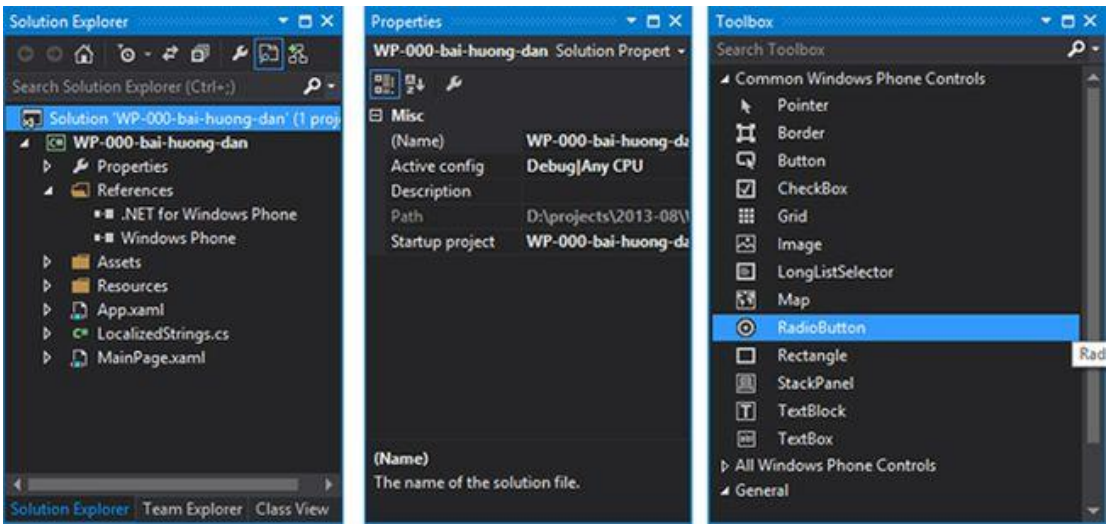
Beberapa komponen kerja program *visual basic 2010* telah ditampilkan sebagai tampilan *standard*. Masih banyak lagi *komponen* yang masih tersembunyi sehingga memerlukan perintah tertentu untuk menampilkannya. Kita dapat

mengatur *komponen* di dalam program *visual basic 2010* sesuai dengan yang kita butuhkan. Berikut ini adalah beberapa *komponen* kerja dari *visual basic 2010* adalah :

a. Toolbox

Toolbox adalah sebuah *panel* yang menampung tombol-tombol yang berguna untuk membuat suatu *desain* mulai dari tombol *label, pointer, button*, dan lain-lain. Berikut ini adalah gambaran *toolbox* pada *visual basic 2010* :

Berikut ini adalah *table* yang berisi nama tombol yang terdapat didalam *toolbox* beserta fungsinya.



Gambar 4. Tampilan *Toolbox*

Sumber : (Safik : 2012 : 2)

Table 1. *Toolbox*

Nama tombol	fungsi
Pointer	Memilih, mengatur ukuran dan memindahkan posisi yang terpasang di bagian *form*.

Bindingsources	Untuk meng*koneksi*kan program ke *database*
Label	Menampilkan *teks*, dimana pengguna program tidak bisa mengubah *teks* tersebut
Groupbox	Untuk mengelompokkan *item* yang ada di *form*
Checkbox	Membuat kotak periksa, dimana pengguna program dapat memilih sekaligus
Listbox	Membuat daftar pilihan
Timer	Membuat *control* waktu dan *interval* yang diperlukan
Image	Menampilkan gambar pada *form* dalam format *bitmap*, *icone*, atau *metafile*
Picturebox	Menampilkan gambar dari sebuah *file*
Textbox	Membuat *teks*, dimana *teks* tersebut dapat diubah oleh pembuat program
Button	Membuat tombol perintah
Combobox	Menambahkan *control* kotak *combo* yang merupakan *control* gabungan antara *textbox* dan *listbox*

Sumber : (Safik : 2012 : 2).

7. *Unified Modeling Language* (*UML*)

a) Pengenalan *UML*

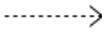
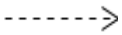
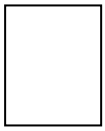
Unified Modelling Language (*UML*) adalah suatu alat untuk mem*visualisasi*kan dan men*dokumentasi*kan hasil *analisa* dan *desain* yang berisi *sintak* dalam me*model*kan sistem secara *visual* (*Haviluddin : 2011 : 1*). Banyak orang yang telah membuat bahasa pe*model*an pembangunan perangkat lunak sesuai dengan teknologi pemrograman yang berkembang pada saat itu, misalnya yang sempat berkembang dan digunakan oleh banyak pihak adalah *Data Flow Diagram* (*DFD*) untuk me*model*kan perangkat lunak yang menggunakan pemrograman *prosedural* atau struktur, kemudian juga ada *State Transition Diagram* (*STD*) yang digunakan untuk me*model*kan *real time* (waktu nyata).

Pada perkembangan teknik pemrograman ber*orientasi objek*, muncullah sebuah *standarisasi* bahasa pe*model*an untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemrograman *berorientasi objek*, yaitu *Unified Modeling Language* (*UML*).

b) *Use Case Diagram*

Diagram yang menggambarkan *actor*, *use case* dan *relasi*nya sebagai suatu urutan tindakan yang memberikan nilai terukur untuk *aktor*. Sebuah *use case* digambarkan sebagai *elips horizontal* dalam suatu diagram *UML use case* (*Haviluddin : 2011 : 4*).

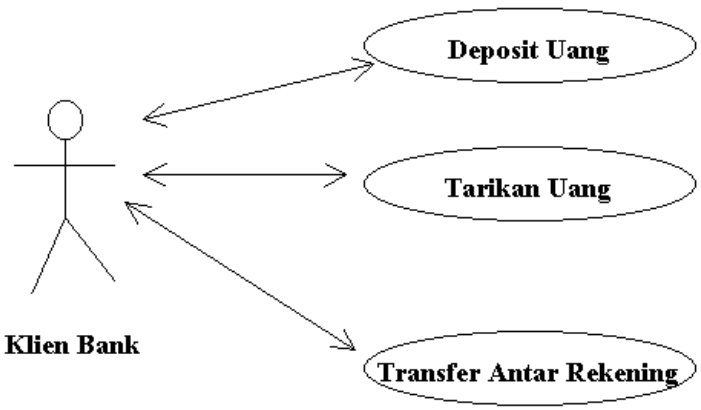
Tabel 2. Simbol *Use Case Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		*Actor*	Menspesifikasikan himpuan peran yang pengguna mainkan ketika ber*interaksi* dengan *use case*.
2		*Dependency*	Hubungan dimana perubahan yang terjadi pada suatu *elemen* mandiri *(independent)* akan mempengaruhi *elemen* yang bergantung padanya *elemen* yang tidak mandiri (*independent*).
3		*Generalization*	Hubungan dimana objek anak (*descendent*) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (*ancestor*).
4		*Include*	Menspesifikasikan bahwa *use case* sumber secara *eksplisit*.
5		*Extend*	Menspesifikasikan bahwa *use case* target memperluas perilaku dari *use case* sumber pada suatu titik yang diberikan.
6		*Association*	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		*System*	Menspesifikasikan paket yang menampilkan sistem secara terbatas.

8		*Use Case*	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
9		*Collaboration*	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan prilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
10		*Note*	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

Sumber : (*Gellysa Urva, 94 : 2015*)

Contoh *Use Case* Diagram :



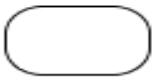
Gambar 5. Contoh *Use Case Diagram*

Sumber : (*Haviluddin : 2011 : 4*)

c) *Activity Diagram*

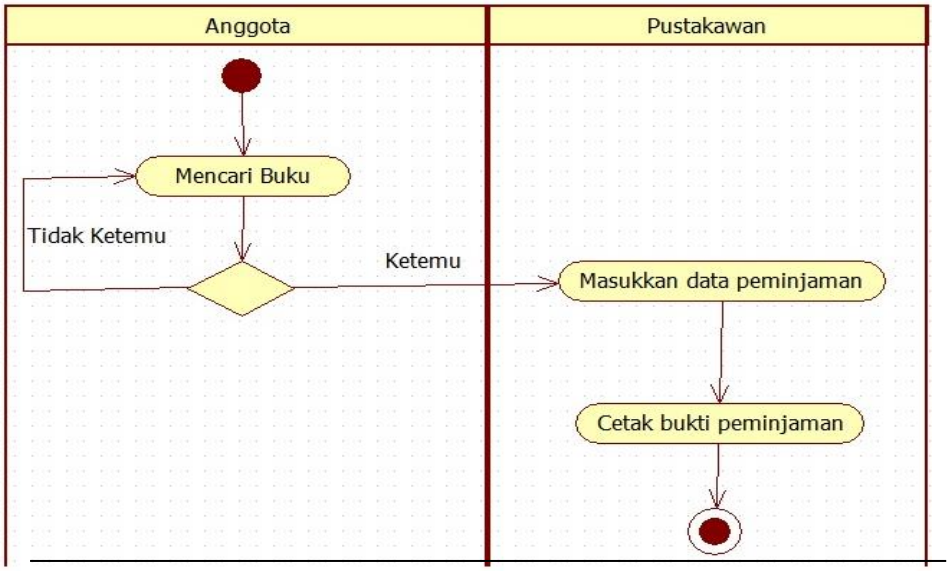
Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau *menu* yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan *aktor*, jadi aktivitas yang dapat dilakukan oleh sistem.

Tabel 3. Simbol *Activity Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		*Activity*	Memperlihatkan bagaimana masing-masing kelas antarmuka saling ber*interaksi* satu sama lain
2		*Action*	*State* dari sistem yang mencerminkan *eksekusi* dari suatu aksi
3		*Initial Node*	Bagaimana objek dibentuk atau diawali.
4		*Actifity Final Node*	Bagaimana objek dibentuk dan dihancurkan
5		*Fork Node*	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

Sumber : (*Gellysa Urva, 94 : 2015*)

Contoh *Activity* Diagram :



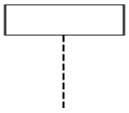
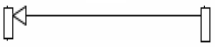
Gambar 6. Contoh *Activity Diagram*

Sumber : (*Gellysa Urva, 94 : 2015*)

d) *Sequence Diagram*

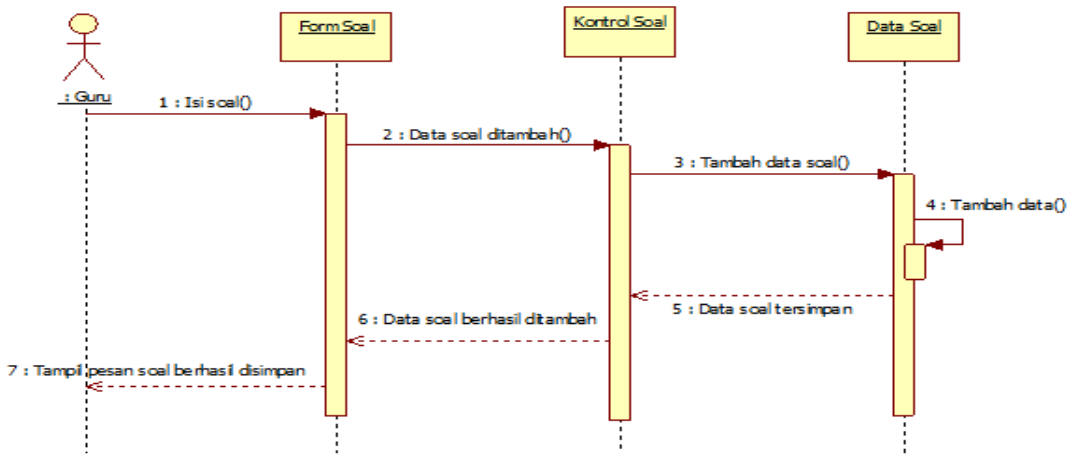
Diagram *sekuen* menggambarkan kelakuan objek pada *use case* dengan men*deskripsi*kan waktu hidup *objek* dan *message* yang dikirimkan dan diterima antar *objek*. Oleh karena itu untuk menggambar diagram *sekuen* maka harus diketahui *objek-objek* yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang di*instansiasi* menjadi *objek* itu. Membuat *diagram sekuen* juga dibutuhkan untuk melihat skenario yang ada pada *use case*.

Tabel 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 : (*Gellysa Urva, 95 : 2015*)

Contoh *Squence* Diagram :



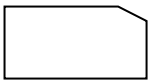
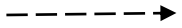
Gambar 7. Contoh *Squence Diagram*

Sumber : (*Gellysa Urva, 95 : 2015*)

e) *Class Diagram*

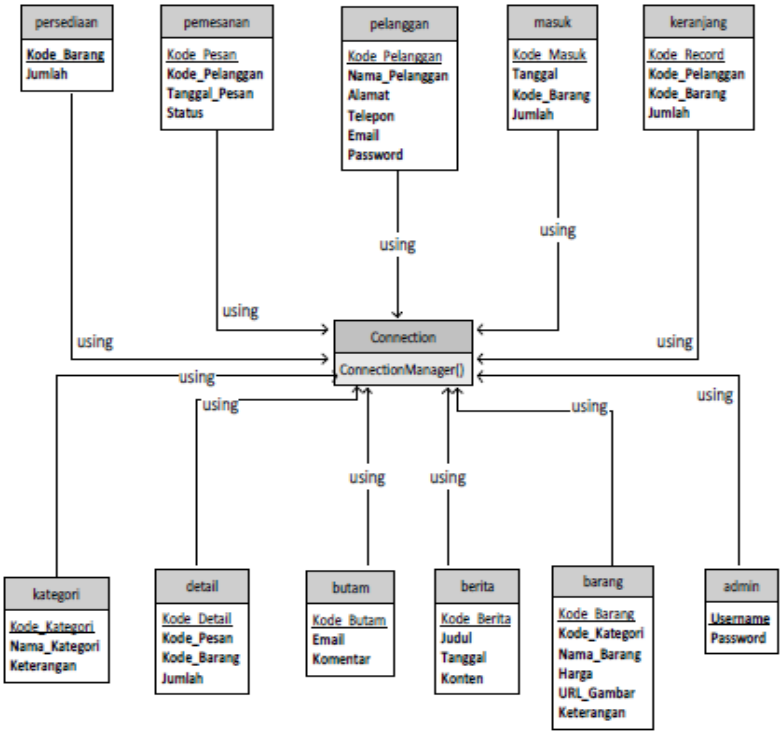
Class diagram menggambarkan struktur *statis* dari kelas dalam sistem anda dan menggambarkan *atribut*, operasi dan hubungan antara kelas. *Class* diagram membantu dalam mem*visualisasi*kan struktur kelas-kelas dari suatu sistem dan merupakan tipe diagram yang paling banyak dipakai. Selama tahap *desain*, *class* diagram berperan dalam menangkap struktur dari semua kelas yang membentuk arsitektur sistem yang dibuat.

Tabel 5. Simbol *Class Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		*note*	*Elemen* fisik yang *eksis* saat *aplikasi* dijalankan dan mencerminkan suatu sumber daya *komputasi*
2		*dependency*	Hubungan dimana perubahan yang terjadi pada suatu *elemen* mandiri akan mempengaruhi *elemen* yang bergantung padanya
3		*extend*	Men*spesifikasi*kan bahwa use case target memperluas prilaku dari use case sumber pada suatu titik yang diberikan.

Sumber : (*Gellysa Urva, 95 : 2015*)

Contoh *Class* Diagram :



Gambar 8. Contoh *Class Diagram*

Sumber : (*Gellysa Urva, 95 : 2015*)

BAB III

ANALISA DAN PERANCANGAN SISTEM

1. Analisa Sistem Yang Berjalan

Dalam materi perkuliahan Keamanan komputer terdapat bab mengenai *enkripsi*. Salah satu bentuk *enkripsi* adalah menggunakan metode *chipertext*. Untuk mendapatkan hasil *teks* yang diubah (*ciphertext),* menggunakan angka dan *tabel* untuk *konversi*. Penggunaan angka jauh lebih sulit dibandingkan dengan menggunakan *tabel*. *Algortima Vernam Cipher* nantinya akan menganalisa langkah-langkah kerja *algoritma kriptografi Vernam Cipher* tersebut, sehingga nantinya *algoritma kriptografi* yang penulis bangun akan memiliki tingkat kesulitan yang lebih tinggi untuk dipecahkan dibandingkan *algoritma kriptografi Vernam Cipher*.

2. Proses *Enkripsi Vernam Chiper*

Adapun *algoritma enkripsi Vernam Cipher* dalam bentuk *psedocode* dibawah ini :

Langkah 1 : *Input Plaintext*

Langkah 2 : *Input* Kunci

Langkah 3 : Ubah Setiap karakter pada *Plaintext* kedalam bentuk *Ascii Code*

I=0

Jum = LEN(*Plaintext*)

For i=1 to jum

P(i) = Asc(substr(*Plaintext*, 1,i))

Next i

Langkah 4 : Ubah Setiap karakter pada Kunci kedalam bentuk *Ascii Code*

I=0

Jum = LEN(Kunci)

For i=1 to jum

K(i) = Asc(substr(Kunci, 1,i))

Next i

Langkah 5 : Lakukan *Enkripsi* dengan rumus

I=0

Jum = LEN(*Plainteks*)

For i=1 to jum

C(i) = P(i) *XOR* (K(i)

Next i

Langkah 6 : Ubah *Ascii Ciphertext* kedalam bentuk karakter

I=0

Jum = LEN(*Ciphertext*)

For i=1 to jum

Karakter_C(i) = chr(substr(C(i), 1,i))

Next i

3. Proses *Dekripsi Vernam Chiper*

Adapun *algoritma dekripsi Vernam Cipher* dalam bentuk *psedocode* dibawah ini :

Langkah 1 : *Input Cipherteks*

Langkah 2 : *Input* Kunci

Langkah 3 : Ubah Setiap karakter pada *Ciphertext* kedalam bentuk *Ascii Code*

```
I=0
Jum = LEN(Ciphertex )
For i=1 to jum
        C(i) = Asc(substr(Ciphertext , 1,i))
Next i
```

Langkah 4 : Ubah Setiap karakter pada Kunci kedalam bentuk *Ascii Code*

```
I=0
Jum = LEN(Kunci)
For i=1 to jum
        K(i) = Asc(substr(Kunci, 1,i))
Next i
```

Langkah 5 : Lakukan *Enkripsi* dengan rumus

```
I=0
Jum = LEN(Ciphertext)
For i=1 to jum
```

P(i) = C(i) *XOR* (K(i)

Next i

Langkah 6 : Ubah *Ascii Plaintext* kedalam bentuk karakter

I=0

Jum = LEN(*Plaintext*)

For i=1 to jum

Karakter_P(i) = chr(substr(*Plaintext* ,1,i))

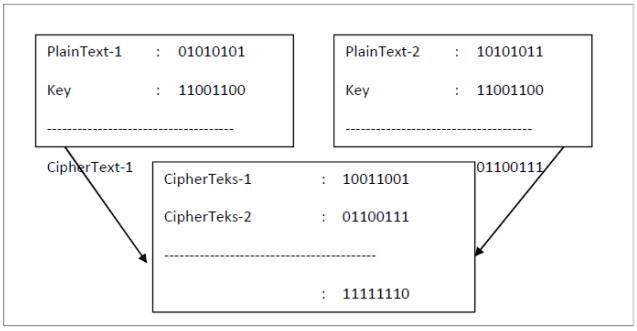
Next i

4. Analisis Kelemahan *Algoritma XOR* dan *Tabel ASCII*

Adapun kelemahan terlihat pada kotak yang ada di *psedocode* diatas, kelemahan *algoritma Vernam Cipher* ini terletak pada pemakaian *XOR* dalam melakukan *enkripsi* dan *dekripsi* antara *plaintext* dan kunci (*Agustanti, 2010*)

P(i) = C(i) *XOR* K(i)

Dimana jika diasumsikan A berhasil menyadap 2 buah *Ciphertext* berbeda dengan kunci yang sama, A kemudian meng *XOR* kan kedua *Ciphertext* tersebut, jika berhasil mengetahui *plainteks* dari *Ciphertext* tersebut maka akan dengan mudah menemukan *plaintext* yang lain tanpa perlu mengetahui rangkaian kuncinya seperti contoh dibawah ini :



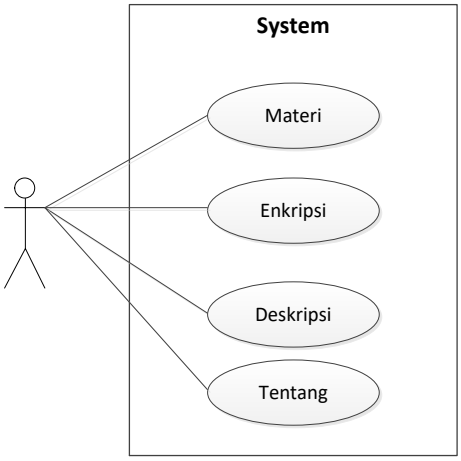
Gambar 9. Contoh Hasil *XOR* Sehingga mendapatkan *Plainteks*

5. Perancangan *Berorientasi* Objek

Tujuan dari perancangan *berorientasi* ojbek ini memungkinkan adanya komunikasi yang lebih berkualitas antara pengguna, pengembang penganalisis, *tester*, manajer dan siapapun yang terlibat dalam proyek pengembangan sistem informasi.

a. *Use case* Diagram

Berikut adalah *use case diagram* yang menggambarkan kegiatan.



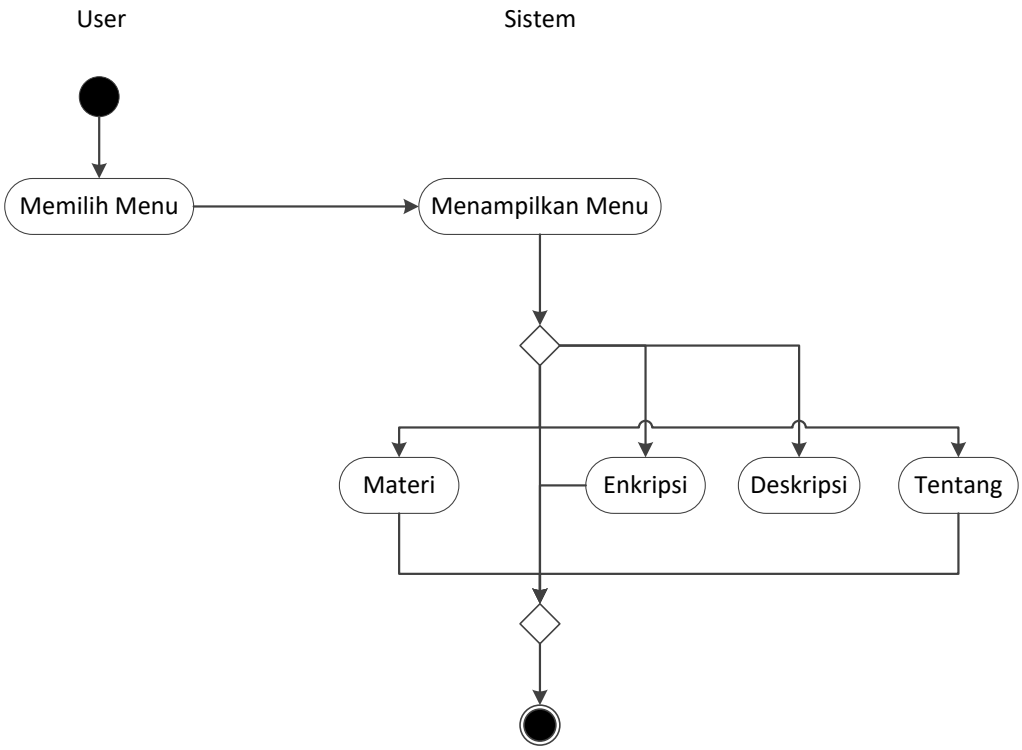
Gambar 10. *Use Case Diagram*

Keterangan :

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

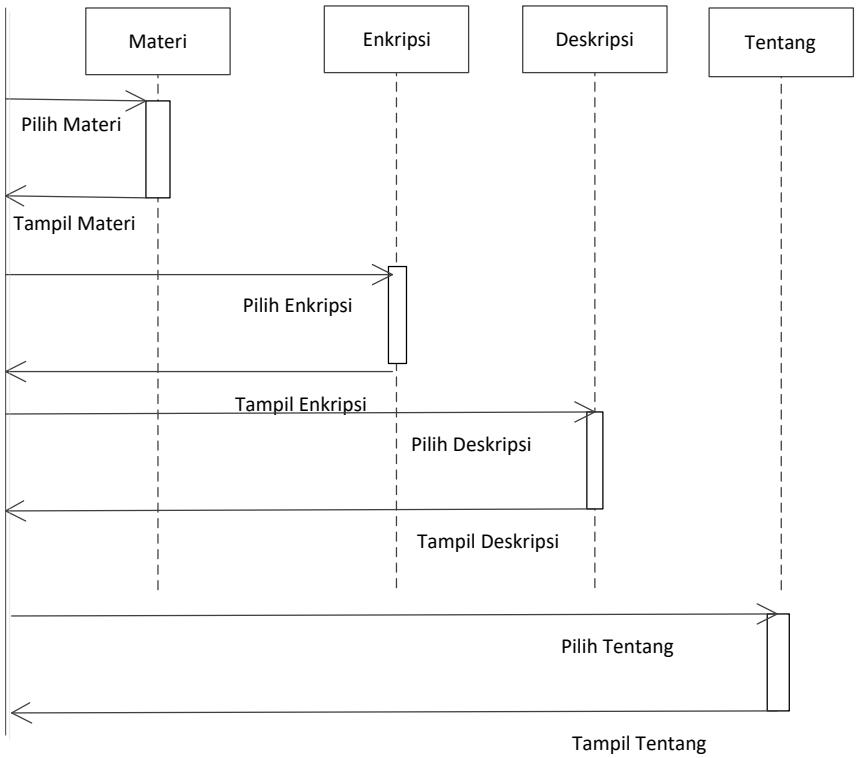
b. Pembuatan *Activity* Diagram

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



Gambar 11. ***Activity Diagram***

c. *Sequence* Diagram



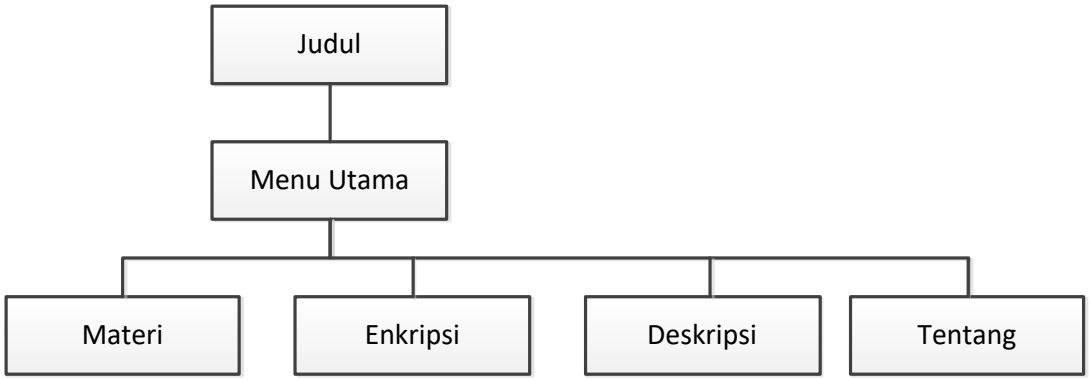
Gambar 12. *Sequence Diagram*

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.

6. Struktur Program

Struktur program mempresentasikan organisasi komponen program (modul) serta meng*implementasi*kann suatu hirarki kontrol. Hirearki kontrol tidak meng*implementasi*kan aspek *prosedural* dari perangkat lunak seperti urutan proses, kejadian atau urutan dari keputusan atau perulangan operasi.



Gambar 13. Struktur *Navigasi Enkripsi*

7. Perancangan Antarmuka

A. Rancangan Halaman Judul

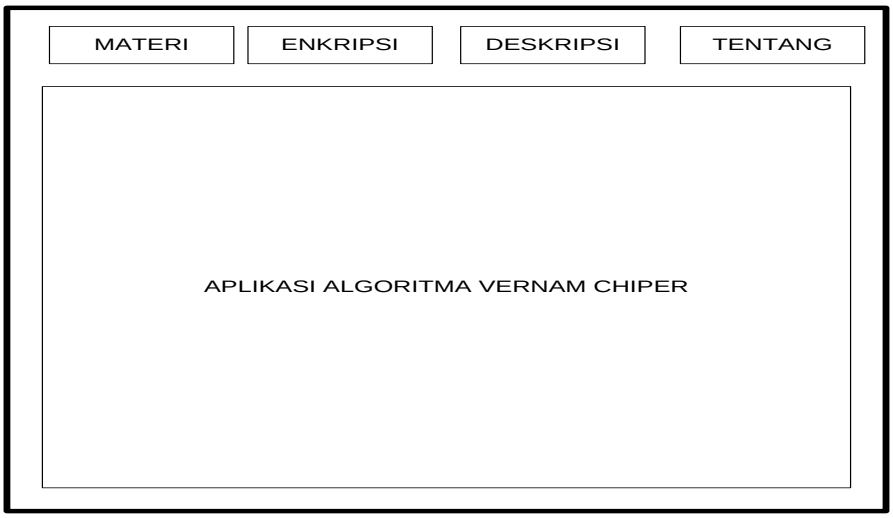
Halaman judul merupakan halaman yang pertama muncul pada saat program dijalankan

Gambar 14. Rancangan Halaman Judul

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

B. Rancangan Halaman Menu Utama

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



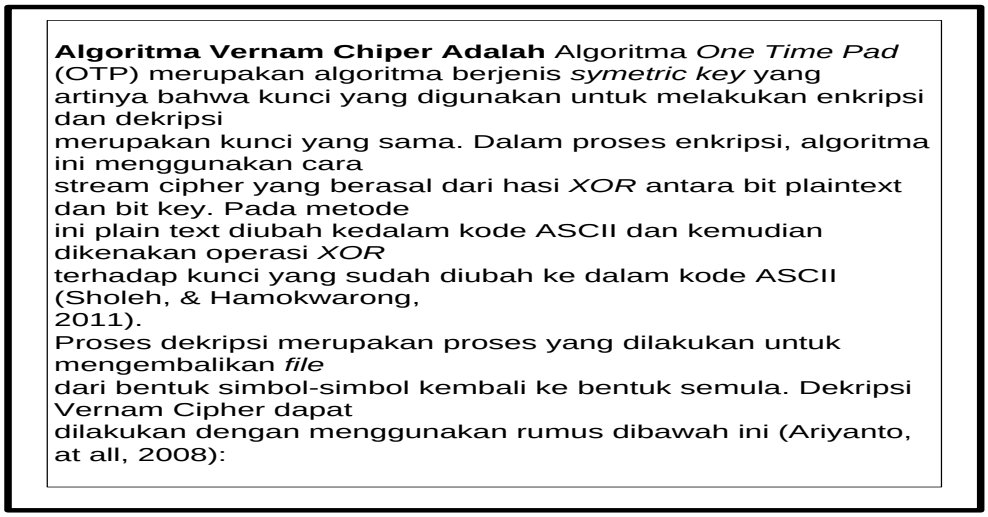
Gambar 15. Rancangan Halaman *Menu* Utama

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

a. Tombol Materi berfungsi untuk menghubungkan pengguna ke *form* materi.
b. Tombol *Enkripsi* berfungsi untuk menghubungkan pengguna ke *form Enkripsi*.
c. Tombol *Deskripsi* berfungsi untuk menampilkan *form Deskripsi*.
d. Tombol Tentang berfungsi untuk menghubungkan pengguna ke *form* tentang.
e. Tombol Keluar berfungsi untuk keluar dari program.

C. Rancangan Halaman Materi

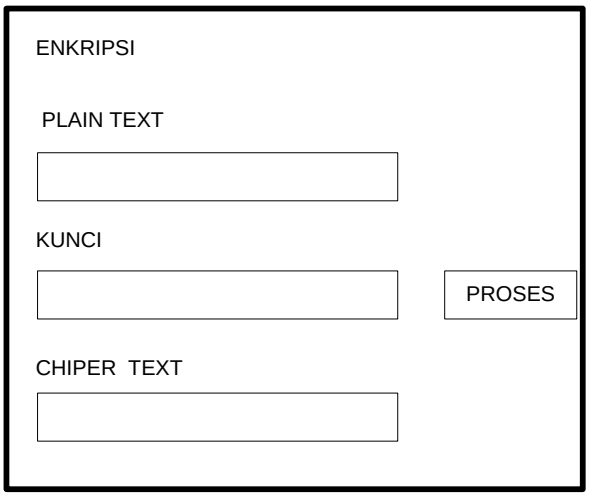
Form ini digunakan untuk menjelaskan cara kerja penyandian, dimulai dari *plaintext* kemudian kunci yang di*konversi*kan dalam bentuk angka. Setelah itu dilakukan proses penjumlahan dan jika hasil penjumlahan maka akan dikurangi 6 lalu hasilnya akan dikembalikan lagi ke dalam bentuk huruf.



Gambar 16. Rancangan Halaman Materi

D. Rancangan Halaman *Enkripsi*

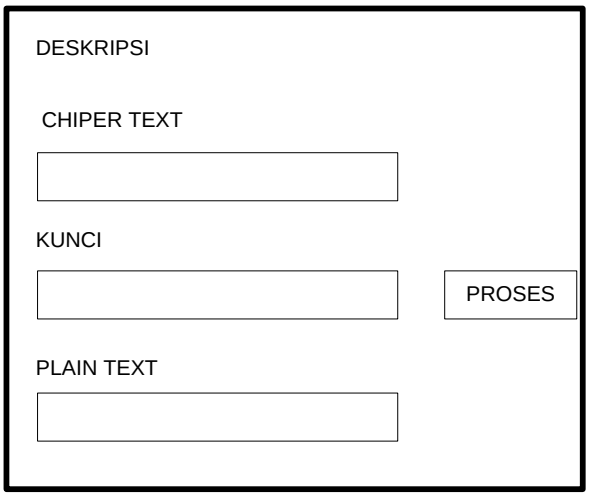
Berisi penjelasan mengenai *Enkripsi*. Pengguna memasukkan tulisan asli atau *plaintext* ke dalam tombol masukan *plaintext* kemudian dimasukkan juga kunci. Setelah itu, ditekan tombol Proses *Enkripsi* yang kemudian akan menampilkan *ciphertext* atau tulisan yang telah disandikan.



Gambar 17. Rancangan Halaman *Enkripsi*

E. Rancangan Halaman *Dekripsi*

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 18. Rancangan Halaman *Deskripsi*

Pada gambar di atas terdapat kotak *input Dekripsi* 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*.

F. Rancangan Halaman *About*

Berisi mengenai *versi* program dan pembuat program.

Gambar 19. *Menu About*

BAB IV

HASIL DAN PEMBAHASAN

1. Implementasi Sistem

Tahap *implementasi* sistem merupakan tahap dimana *aplikasi* yang telah dirancang di jalankan. Tahap ini menunjukkan apakah setiap proses dapat berjalan dengan baik dan mampu memberikan hasil yang diharapkan. Proses perancangan *aplikasi* menggunakan *Visual Basic NET 2010* di tampilkan dalam bentuk *form-form* yang menjadi sarana bagi pengguna untuk melakukan proses *implementasi*.

2. Pengujian Sistem

Pengujian sistem dilakukan untuk menunjukkan apakah sistem yang telah dirancang dapat berjalan sesuai harapan. Selain itu tujuan pengujian adalah untuk dapat menemukan kesalahan fungsi pada *aplikasi* yang dibangun dan memperbaikinya.

Pengujian dilakukan dengan memasukkan karakter atau huruf dari *file* ber*format* *.*txt* selanjutnya diproses oleh *aplikasi* apakah *aplikasi* tersebut dapat memberikan hasil yang sesuai. Proses yang akan dilakukan pengujian dalam *aplikasi* ini adalah simulasi pengiriman pesan dengan menggunakan metode *algoritma vernam* antara pengirim kepada penerima dengan kunci yang di miliki masing-masing pihak tanpa perlu bertukar kunci tunggal hingga pada akhirnya pesan asli yang dikirim kan oleh pengirim dapat dibaca oleh penerima .

a. Tampilan Awal / *Home*

Tampilan pada gambar 20 merupakan tampilan awal ketika *aplikasi* dijalankan. Pada *form* ini pengguna dapat memilih untuk membuka beberapa *form* lainnya seperti tombol tentang yang akan mengarahkan pengguna menuju *form* yang menjelaskan profil *aplikasi* ini, tombol *read me!* Yang akan mengarahkan pengguna ke *form* yang menjelaskan tata cara penggunaan dari *aplikasi* ini.

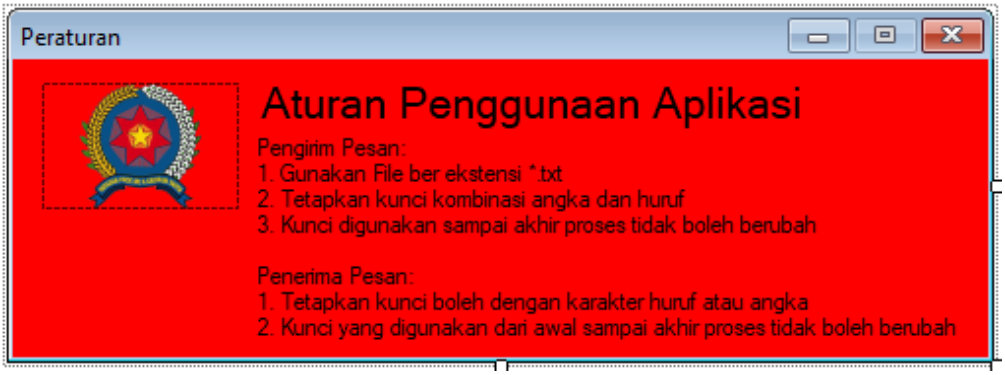
Gambar 20. Tampilan Awal / *Home*

Keterangan:

1. Mulai : Proses Untuk melanjutkan ke *Form* selanjutnya yaitu *form menu* utama.
2. *Read Me* : Berfungsi untuk menampilkan dan menjelaskan proses *algortima vernam.*
3. Tentang : Berfungsi untuk menampilkan tentang pembuat *aplikasi* ini

b. Tampilan Aturan Penggunaan *Aplikasi*

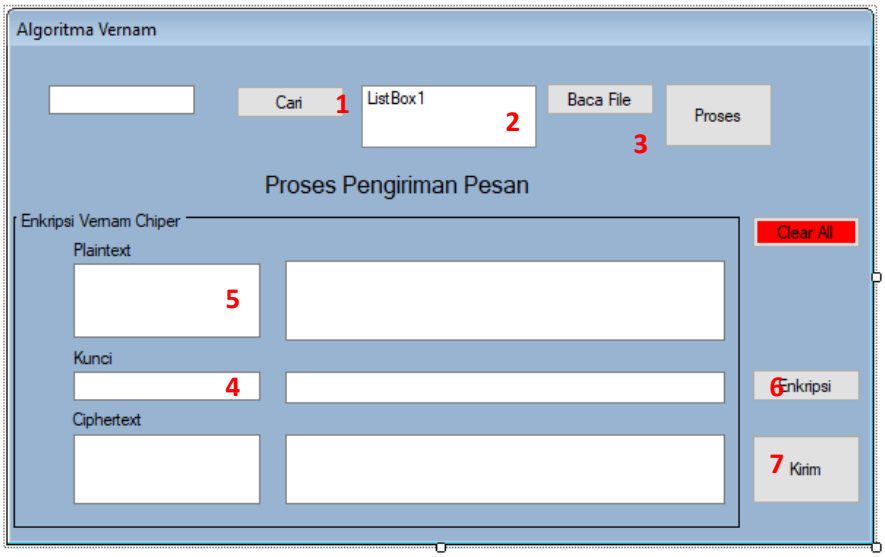
Tampilan aturan penggunaan *aplikasi* merupakan tampilan halaman atau *form* yang berisi tentang tata cara penggunaan *aplikasi* yang dijalankan. Padahal halaman tersebut di jelaskan apa-apa saja yang menjadi kewajiban bagi pengirim dan penerima saat menjalankan simulasi *algoritma vernam*.



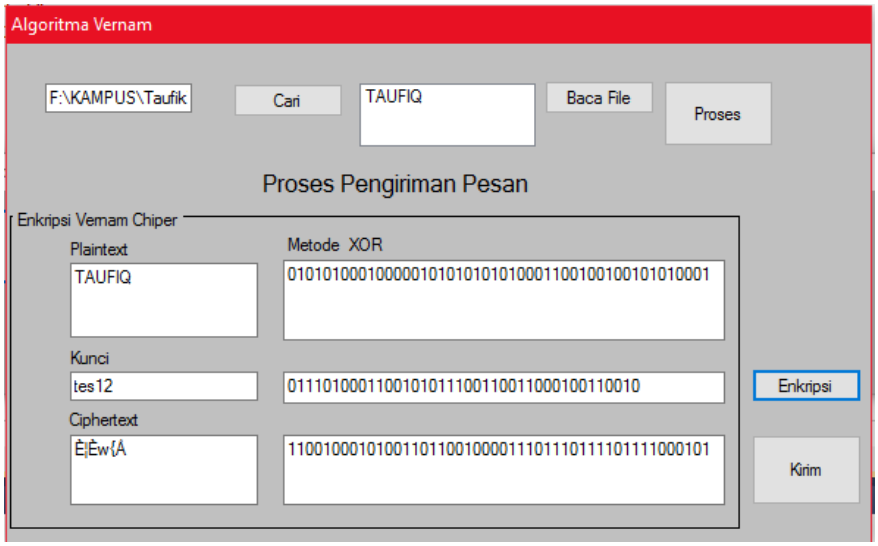
Gambar 21. Tampilan Aturan Penggunaan *Aplikasi*

c. Tampilan Halaman *Enkripsi Algoritma Vernam*

Tampilan berikut merupakan tampilan utama pada *aplikasi* ini. *algoritma vernam* merupakan protokol yang menjamin tidak adanya pertukaran kunci antara pihak-pihak yang melakukan *enkripsi* dan *dekripsi*. Kedua belah pihak menggunakan kunci mereka masing-masing untuk meng*enkripsi* pesan dan kemudian untuk men*dekripsi* pesan tanpa perlu mengetahui kunci yang lainnya



Gambar 22. Tampilan Halaman Utama *Algoritma Vernam*



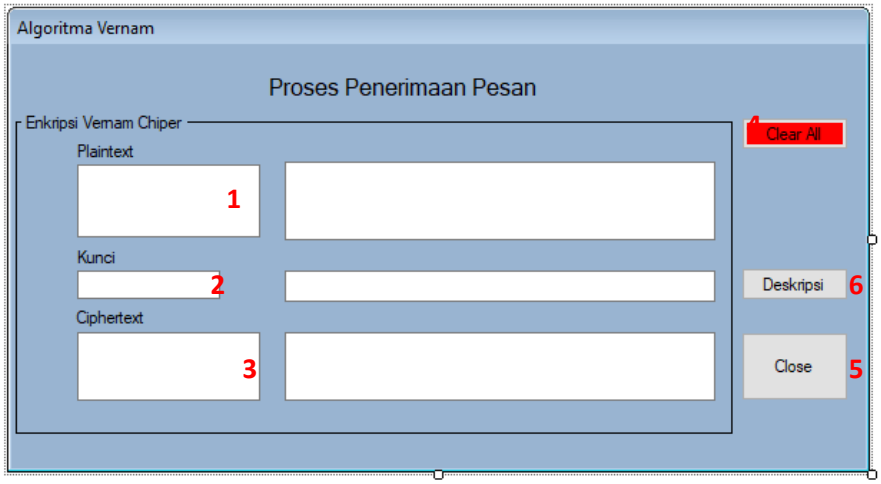
Gambar 23. Proses Tampilan Halaman Utama *Algoritma Vernam*

Keterangan:

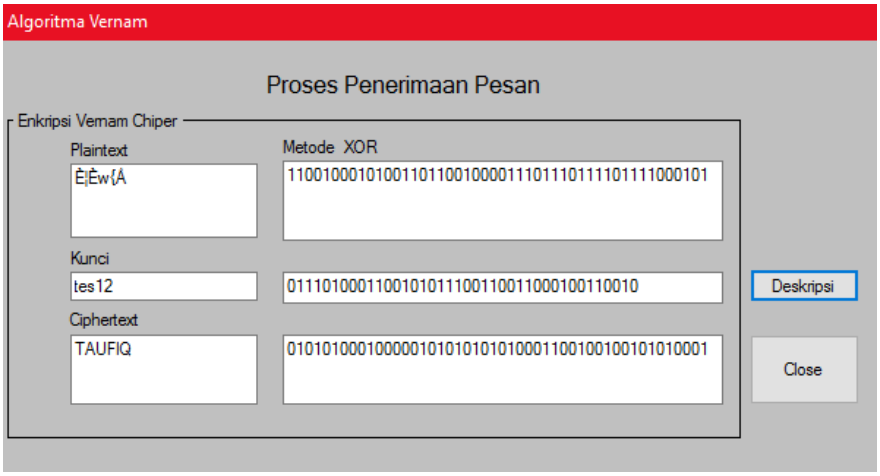
1. Cari : Berfungsi untuk membuka *file* pesan yang ber*format* **.txt*
2. *Lisboxt1* : Berfungsi untuk memilih *text* atau pesan yang akan diproses menggunakan *algoritma vernam chiper*.

3. Baca File : Berfungsi untuk memindahkan *text* atau pesan yang ada pada *listbox* ke *plaintext*.
4. Kunci : Berfungsi untuk memberikan sandi dan mengubah pesan asli menjadi acak.
5. *Chipertext* : Berisikan pesan yang akan dikirim oleh penerima pesan.
6. *Enkripsi* : Proses acak dengan menggunakan metode *XOR* menggunakan *Alogritma Vernam*.
7. Kirim : Berfungsi untuk mengirimkan pesan *Chipertext* ke Penerima Pesan.

d. Tampilan Halaman *Dekripsi Algoritma Vernam*



Gambar 24. Tampilan Halaman Utama *Algoritma Vernam*



Gambar 25. Proses Tampilan Halaman Utama *Algoritma Vernam*

Keterangan:

1. *Plaintext* : Pesan yang diterima dari pengirim pesan.
2. Kunci : Berfungsi untuk membuka pesan asli dari *plaintext* yang dikirimkan oleh penerima pesan.
3. *Chipertext* : Berfungsi untuk menampilkan pesan asli yang dikirimkan oleh si pengirim pesan.
4. *Clear All* : Berfungsi untuk mengosongkan seluruh *textbox* dan *listbox*
5. *Close* : Berfungsi untuk menutup *Form Dekripsi* Pesan
6. *Enkripsi* : Proses acak dengan menggunakan metode *XOR* menggunakan untuk membuka pesan.

3. Proses Perhitungan Pada Sistem

Saya memiliki sebuah *plaintext* yaitu TAUFIK dan memiliki sebuah kunci yaitu CRASH (ingat panjang kunci harus sama dengan *plaintext* dan sebaiknya tidak ada karakter yang diulang).

Pertama kita harus mendapatkan kode *ASCII* dari *plaintext* kemudian diubah ke bentuk *biner*

Karakter	ASCII	Notasi biner
T	84	01010100
A	65	01000001
U	85	01010101
F	70	01000110
I	73	01001001
K	75	01001011

Hal yang sama juga harus dilakukan pada kunci yang dipilih.

Karakter	ASCII	Notasi biner
C	67	0100 0011
R	82	0101 0010
A	65	0100 0001
S	83	0101 0011
H	72	0100 1000

Setelah itu masing-masing karakter di *XOR*-kan dengan *Key*

```
T=01010100 A= 01000001 U= 01010101 F= 01000110 I=
01001001 K= 01001011
 C=01000011 R= 01010010 A= 01000001  S=01010011 H=01001000

XOR ------------------------------------------------------------
--
Cipher: 10000000 01010010 00000001 11001000 00000100 1110000
10010001 01000000 00110001 01111101
------------------------------------------------------------------
--
ASCII : —  “ – ™ ‘ Ž
```

Proses *dekripsi* pesan juga melakukan operasi yang sama yaitu *XOR* antara *Cipher* dengan *key*.

```
Cipher: 10000000 01010010 00000001 11001000 00000100 1110000
10010001 01000000 00110001 01111101
Key:    0100 0011   0101 0010   0100 0001   0101 0011      0100
1000
XOR ------------------------------------------------------------
--
Plain : 01010100 01000001 01010101 01000110 01001001 01001011
------------------------------------------------------------------
--
ASCII :   T A U F I K
```

BAB V

PENUTUP

1. Kesimpulan

Berdasarkan pembahasan dalam perancangan Penerapan *Algoritma Vernam Cipher* dalam Pengamanan Data, maka dapat diambil kesimpulan sebagai berikut :

1. Perangkat lunak ini dirancang untuk menampilkan simulasi pengiriman pesan berekstensi **.txt* antara pengirim dan penerima.
2. Penggunaan *Algoritma Vernam* sangat baik digunakan untuk proses pengamanan data.
3. Penggunaan kunci sulit ditebak dikarenakan menggunakan *text to binary*.
4. Kemungkinan bocornya kunci saat proses pertukaran informasi kunci tunggal dapat dihindari.

2. Saran

Adapun saran-saran yang dapat dilakukan penelitian ataupun pengembangan selanjutnya adalah sebagai berikut:

1. Perangkat lunak ini dapat dikembangkan dengan menggunakan kombinasi metode-metode lain.
2. Perangkat lunak ini dapat dikembangkan dan terhubung ke jaringan sehingga dapat dijalankan di lebih dari satu computer.
3. Perangkat lunak ini dapat dikembangkan menggunakan algoritma-algoritma lain yang lebih kompleks.

DAFTAR PUSTAKA

Cheddad, J. Condell, K. Curran and *P. Mc Kevitt* , "*Digital Image Steganography*: *Survey and Analysis of Current Methods*," *International Journal of Signal Processing, vol*. 90, *no*. 3, pp. 727-752, 2010.

Rachmawanto and C. A. Sari, "Keamanan *File* Menggunakan Teknik *Kriptografi Shift Cipher*," Jurnal *Techno. Com*, *vol*. 14, *no*. 4, pp. 329-335, 2015.

Sari, E. H. Rachmawanto, Y. P. Astuti and L. Umaroh, "*Optimasi* Penyandian *File Kriptografi Shift Cipher*," *in Prosiding Sendi_U* 2016, Semarang, 2016.

Rachmawanto, C. A. Sari, Y. P. Astuti and L. Umaroh, "*Kriptografi* Dengan *Algoritma Vernam cipher* Untuk Keamanan Data," *in Prosiding Sendi_U* ke 2 Tahun 2016, Semarang, 2016.

Kromodimoeljo, Teori dan *Aplikasi Kriptografi*, Jakarta: SPK IT Consulting, 2009.

Ariyus, Pengantar Ilmu Kriptografi: Teori, Analisi dan Implementasi, Yogyakarta: Andi, 2008.

Sadikin, Kriptografi Untuk Keamanan Jaringan, Yogyakarta: Andi, 2012.

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.

Mariance, U. C. (2018). Analisa dan Perancangan Media Promosi dan Pemasaran Berbasis Web Menggunakan Work System Framework (Studi Kasus di Toko Mandiri Prabot Kota Medan). Jurnal Ilmiah Core IT: Community Research Information Technology, 6(1).

Marlina, L., Muslim, M., Siahaan, A. U., & Utama, P. (2016). Data Mining Classification Comparison (Naïve Bayes and C4. 5 Algorithms). Int. J. Eng. Trends Technol, 38(7), 380-383.

Mayasari, Nova. "Comparison of Support Vector Machine and Decision Tree in Predicting On-Time Graduation (Case Study: Universitas Pembangunan Panca Budi)." Int. J. Recent Trends Eng. Res 2.12 (2016): 140-151.

Muttaqin, Muhammad. "Analisa pemanfaatan sistem informasi e-office pada universitas pembangunan panca budi medan dengan menggunakan metode utaut." Jurnal Teknik dan Informatika 5.1 (2018): 40-43.

Perwitasari, I. D. (2018). Teknik Marker Based Tracking Augmented Reality untuk Visualisasi Anatomi Organ Tubuh Manusia Berbasis Android. INTECOMS: Journal of Information Technology and Computer Science, 1(1), 8-18.

Puspita, Khairani, and Purwa Hasan Putra. "Penerapan Metode Simple Additive Weighting (SAW) Dalam Menentukan Pendirian Lokasi Gramedia Di Sumatera Utara." Seminar Nasional Teknologi Informasi Dan Multimedia, ISSN. 2015.

Putera, A., Siahaan, U., & Rahim, R. (2016). Dynamic key matrix of hill cipher using genetic algorithm. Int. J. Secur. Its Appl, 10(8), 173-180.

Putra, Randi Rian, and Cendra Wadisman. "Implementasi Data Mining Pemilihan Pelanggan Potensial Menggunakan Algoritma K Means." INTECOMS: Journal of Information Technology and Computer Science 1.1 (2018): 72-77.

Putri, R. E., & Siahaan, A. (2017). Examination of document similarity using Rabin-Karp algorithm. International Journal of Recent Trends in Engineering & Research, 3(8), 196-201.

Rahim, R. (2018, October). A Novelty Once Methode Power System Policies Based On SCS (Solar Cell System). In International Conference of ASEAN Prespective and Policy (ICAP) (Vol. 1, No. 1, pp. 195-198).

Rizal, Chairul. "Pengaruh Varietas dan Pupuk Petroganik Terhadap Pertumbuhan, Produksi dan Viabilitas Benih Jagung (Zea mays L.)." ETD Unsyiah (2013).

Ruwaida, D., & Kurnia, D. (2018). Rancang Bangun File Transfer Protocol (FTP) dengan Pengamanan Open SSL pada Jaringan VPN Mikrotik di SMK Dwiwarna. CESS (Journal of Computer Engineering, System and Science), 3(1), 45-49.kbar, A. (2018). Pembangunan Model Electronic Government Pemerintahan Desa Menuju Smart Desa. Jurnal Teknik dan Informatika, 5(1), 1-5.

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.