APLIKASI ENKRIPSI DAN DEKRIPSI DENGAN TEKNIK XOR MENGGUNAKAN METODE VERNAM CIPHER

Disusun dan Diajukan Sebagai Salah Satu Persyaratan Untuk Memenuhi Ujian Akhir
Memperoleh Gelar Sarjana Komputer Pada Fakultas Sains Dan Teknologi
Universitas Pembangunan Panca Budi
Medan

SKRIPSI

OLEH

NAMA	:	**MHD. FIRMANSYAH**
N.P.M	:	**1514370629**
PROGRAM STUDI	:	**SISTEM KOMPUTER**

FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS PEMBANGUNAN PANCA BUDI
MEDAN
2020

ABSTRAK

MHD. FIRMANSYAH
APLIKASI ENKRIPSI DAN DEKRIPSI DENGAN TEKNIK XOR MENGGUNAKAN METODE VERNAM CIPHER

Perkembangan teknologi terutama pada sistem pengamanan data dalam menjaga keamanan data informasi telah berkembang pesat. Dalam menjaga keamanan data informasi terdapat cabang ilmu dalam pengembangannya seperti kriptografi. Pada penerapannya dilakukan tidak hanya pada satu teknik keamanan saja, melainkan bisa dilakukan dengan kombinasi dalam keamanan data informasi. Penelitian ini bertujuan untuk membuat sebuah sistem keamanan data dengan mengimplementasikan kriptografi pada pesan teks, isi file dokumen, dan file dokumen dengan melakukan perhitungan algoritma Vernam Cipher. Metode Vernam Cipher merupakan algoritma berjenis symmetric key kunci yang digunakan untuk melakukan enkripsi dan dekripsi yang menggunakan kunci yang sama. Dalam proses enkripsi, algoritma Vernam Cipher menggunakan cara stream cipher dimana cipher berasal dari hasil XOR antara bit plaintext dan bit key, sedangkan permutasi biner dilakukan dengan membalikan kode biner pada setiap karakter Dalam makalah ini akan dibahas, program aplikasi yang dapat melakukan proses kriptografi terhadap suatu file. Proses kriptografi yang terdiri dari enkripsi dan dekripsi akan menggunakan metode Vernam Cipher dan metode permutasi biner.

Kata Kunci : Enkripsi, Dekripsi, XOR, Vernam Cipher, Penyandian Pesan.

DAFTAR ISI

Halaman

BAB IV HASIL DAN PEMBAHASAN

BAB V KESIMPULAN DAN SARAN

DAFTAR PUSTAKA

DAFTAR GAMBAR

Halaman

DAFTAR TABEL

Halaman

KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Allah SWT atas rahmat dan karunia-Nya yang telah diberikan kepada penulis sehingga dapat menyelesaikan Skripsi ini dengan judul: **"APLIKASI ENKRIPSI DAN DEKRIPSI DENGAN TEKNIK XOR MENGGUNAKAN METODE VERNAM CIPHER"**

Dalam penyusunan laporan Skripsi ini penulis menyadari banyak mengalami kesulitan namun berkat bantuan dan dorongan dari berbagai pihak, akhirnya laporan kerja praktek ini dapat juga diselesaikan. Penulis dengan segala kerendahan hati menyampaikan terima kasih kepada:

1. Ayahanda dan Ibunda beserta keluarga yang telah berjasa dalam memberikan dukungan moril dan materil.
2. Bapak H.M. Isa Indrawan, SE.,MM, selaku Rektor Universitas Pembangunanan Panca Budi Medan.
3. Bapak Hamdani, ST.,MT, selaku Dekan Fakultas Sains dan Teknologi Universitas Pembangunan Panca Budi Medan
4. Bapak Eko Hariyanto, S.Kom.,M.Kom, Ketua Program Studi Sistem Komputer Fakultas Sains Dan Teknologi Universitas Pembangunan Panca Budi Medan.
5. Dosen Pembimbing 1 Bapak Andysah Putera Utama Siahaan, S.Kom., Mkom., Ph.D.
6. Dosen Pembimbing 2 Bapak Muhammad Donni Lesmana Siahaan, S.Kom, M.Kom.
7. Seluruh Dosen dan Staf Pegawai Fakultas Sains Dan Teknologi yang telah banyak membantu dalam kelancaran seluruh aktivitas perkuliahan.
8. Teman-teman yang telah memberikan berbagai saran, inspirasi, dorongan, doa, motivasi dan moril maupan materil yang diperlukan sehingga penulis dapat menyelesaikan Skripsi ini.

Akhirnya penulis menyadari sepenuhnya akan Skripsi ini, untuk itu penulis menerima saran kritikan dari semua pihak demi menyempurnakan laporan kerja praktek ini, semoga laporan kerja praktek ini memberi manfaat bagi pembaca dan khususnya penulis sendiri.

Medan, 5 Februari 2020
Penulis,

MHD. FIRMANSYAH
NPM : 1514370629

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Algoritma penyandian data saat ini telah semakin banyak jumlahnya, sejalan dengan berkembangnya ilmu yang mempelajari penyandian data tersebut. Ilmu ini biasa disebut Kriptografi. Dalam kriptografi terdapat bebrapa metode yang cukup penting dalam pengamanan suatu data, untuk menjaga kerahasiaan data salah satunya adalah enkripsi *(encryption).* Enkripsi adalah suatu proses yang dilakukan untuk mengubah pesan asli menjadi *chipertext*. Sedangkan suatu proses yang dilakukan untuk mengubah pesan tersembunyi menjadi pesan asli disebut dekripsi. Pesan biasa atau pesan asli disebut *plaintext* sedangkan pesan yang telah diubah atau disandikan supaya tidak mudah dibaca disebut dengan *chipertext*.

Kriptografi (*cryptographi*) berasal dari BahasaYunani: "cryptos" artinya "*secret*" (rahasia), sedangkan "graphein" artinya "*writing*" (tulisan). Sehingga kriptografi berarti *"secret writing"* (tulisan rahasia). Jadi kriptografi didefinisikan sebagai ilmu dan seni untuk menjaga kerahasiaan pesan dengan cara menyandikannya kebentuk yang tidak dapat dimengerti lagi maknanya. Secara umum kriptografi merupakan teknik pengamanan informasi yang dilakukan dengan cara mengolah informasi awal (*plainteks*) dengan suatu kunci tertentu menggunakan suatu metode enkripsi tertentu sehingga menghasilkan informasi baru (*chipertext*) yang tidak dapat dibaca secara langsung. *Chipertext* tersebut dapat dikembalikan menjadi informasi awal (*plainteks*) melalui proses deskripsi.

1.2 Rumusan Masalah

Berdasarkan dengan latar belakang yang telah dipaparkan di atas maka, penulis dapat merumuskan masalah sebagai berikut:

1. Bagaimana rancangan aplikasi enkripsi dan dekripsi dengan teknik XOR menggunakan algoritma *vernam chiper* ?
2. Bagaimana cara penggunaan aplikasi enkripsi dan deskripsi dengan Teknik XOR menggunakan metode *Venam Cipher*.

1.3 Batasan Masalah

Untuk membahas masalah di atas dalam tugas akhir ini dibatasi oleh hal sebagai berikut:

1. Aplikasi ini dirancang untuk mengamankan text.
2. Teknik yang digunakan adalah Teknik XOR.
3. Penelitian hanya terbatas pada apliaksi enkripsi dan deskripsi.
4. Metode yang digunakan adalah metode *vernam cipher*.

1.4 Tujuan Penelitian

Adapun tujuan penelitian ini adalah sebagai berikut:

1. Sebagai tinjauan tentang metode enkripsi dan dekripsi yang sudah ada dengan algoritma enkripsi yang dibuat akan menjadi acuan dalam pembentukan ide logika dan algoritma yang dikembangkan.
2. Mengetahui tingkat kesulitan dan kelemahan setelah proses enkripsi dijalankan, dan penggunaan perangkat-lunak sebagai aplikasi dari

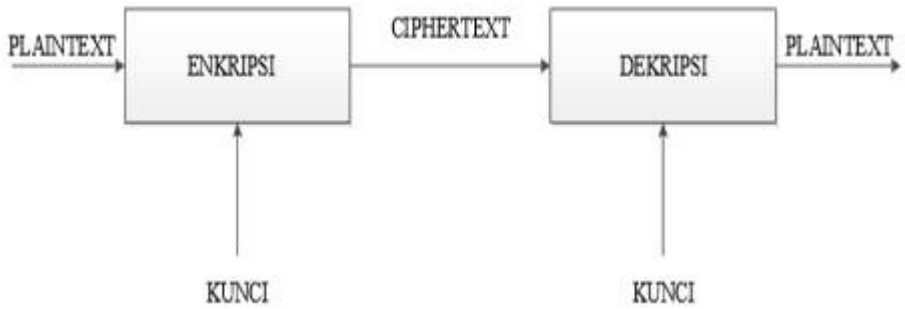
3. algoritma enkripsi dan dekripsi ini, sehingga menghasilkan suatu bentuk program yang dapat dijadikan penyajian data.

BAB II

LANDASAN TEORI

2.1 Kriptografi

Kriptografi (*cryptographi*) berasal dari BahasaYunani: "*cryptos"* artinya "*secret"* (rahasia), sedangkan "*graphein*" artinya "*writing*" (tulisan). Sehingga kriptografi berarti "*secret writing*" (tulisan rahasia). Jadi kriptografi didefinisikan sebagai ilmu dan seni untuk menjaga kerahasiaan pesan dengan cara menyandikannya kebentuk yang tidak dapat dimengerti lagi maknanya. Secara umum kriptografi merupakan teknik pengamanan informasi yang dilakukan dengan cara mengolah informasi awal (plainteks) dengan suatu kunci tertentu menggunakan suatu metode enkripsi tertentu sehingga menghasilkan informasi baru (chipertext) yang tidak dapat dibaca secara langsung. Chipertext tersebut dapat dikembalikan menjadi informasi awal (plainteks) melalui proses deskripsi, (Shohfi Tamam dan Agung Setyabudi, 2016:1).



Gambar 2.1 proses Enkripsi dan Dekripsi

2.1.1 Sejarah Kriptografi

Sejarah kriptografi sebagian besar merupakan sejarah kriptografi klasik, yaitu metode enkripsi yang menggunakan kertas dan pensil atau mungkin dengan bantuan alat mekanik sederhana. Secara umum algoritma kriptografi klasik dikelompokkan menjadi dua kategori, yaitu algoritma transposisi (transposition cipher) dan algoritma substitusi (substitution cipher). Cipher transposisi mengubah susunan huruf-huruf di dalam pesan, sedangkan cipher substitusi mengganti setiap huruf atau kelompok huruf dengan sebuah huruf atau kelompok huruf lain, (Fresly Nandar Pabokory, Indah Fitri Astuti dan Awang Harsa Kridalaksana, 2015:22).

Kriptografi mempunyai sejarah yang sangat menarik dan panjang. Kriptografi sudah digunakan 4000 tahun yang lalu yang diperkenalkan oleh orang-orang Mesir untuk mengirim pesan kepasukan militer yang berada dilapangan dan supaya pesan tersebut tidak terbaca oleh pihak musuh walaupun kurir pembawa pesan tersebut tertangkap oleh musuh.

Pada zaman Romawi kuno dikisahkan pada suatu saat, ketika Julius Caesar ingin mengirimkan satu pesan rahasia kepada seorang Jendral di medan perang. Pesan tersebut harus dikirimkan melalui seorang kurir, tetapi karena pesan tersebut mengandung rahasia, Julius Caesar tidak ingin pesan tersebut terbuka ditengah jalan. Di sini Julius Caesar memikirkan bagaimana mengatasinya yaitu dengan cara mengacak pesan tersebut menjadi suatu pesan yang tidak dapat dipahami oleh siapapun kecuali hanya dapat dipahami oleh Jendralnya saja. Tentu sang Jendral telah diberitahu sebelumnya

bagaimana cara membaca pesan yang teracak tersebut, karena telah mengetahui kuncinya. Yang dilakukan Julius Caesar adalah mengganti semua susunan alfabet dari a,b,c & yaitu a menjadi d, b menjadi e, e menjadi f, dan seterusnya. Sehingga Julius menuliskan kata "saya sekarang & vhndudqi & membacanya".

Dari ilustrasi tersebut beberapa istilah *Cryptography* dipergunakan untuk menandai aktifitas-aktifitas rahasia dengan mengirim pesan. Apa yang dilakukan Julius Caesar dengan cara mengacak pesannya, kita disebut sebagai *encryption* dan pada saat Sang Jendral merapikan pesan yang teracak itu, kita sebut dengan *decryption.* Pesan awal yang belum diacak dan yang telah dirapikan, kita sebut *plaintext* sedangkan pesan yang telah diacak, kita sebut *chipertext.*

Huruf-huruf dengan bentuk tegak akan mempunyai lebar huruf yang lebih kecil dibandingkan dengan huruf-huruf yang melintang, sehingga dengan jumlah huruf yang sama, huruf yang berbentuk melintang akan memakan banyak tempat. Spasi antar huruf juga terlihat berfariasi pada huruf yang melintang dari pada huruf tegak, (Dony Ariyus, 2006).

2.1.2 Tujuan kriptografi

Dari paparan awal dapat dirangkumkan bahwa kriptografi bertujuan untuk member layanan keamanan. Yang dinamakan aspek-aspek keamanan:

1. Kerahasiaan (confidentiality) Adalah layanan yang ditujukan untuk menjaga agar pesan tidak dapat dibaca oleh pihak-pihak yang tidak berhak.
2. Integritas data (data *integrity*) Adalah layanan yang menjamin bahwa pesan masih asli atau belum pernah dimanipulasi selama pengiriman.
3. Otentikasi (*authentication*) Adalah layanan yang berhubungan dengan identifikasi, baik mengidentifikasi kebenaran pihakpihak yang berkomunikasi (*user autehentication*).
4. *Non-repudiation* Adalah layanan untuk menjaga entitas yang berkomunikasi melakukan penyangkalan.

Advanced Encryption Standard (AES) Pada tahun 1997 kontes pemilihan suatu standar algoritma kriptografi baru pengganti DES dimulai dan diikuti oleh 21 peserta dari seluruh dunia. Setelah melewati tahap seleksi yang ketat, pada tahun 1999 hanya tinggal 5 calon yaitu algoritma Serpent (Ross Anderson-University of Cambridge, Eli Biham-Technion, Lars Knudsen-University of California San Diego), MARS (IBM Amerika), Twofish (Bruce Schneier, John Kelsey, dan Niels Ferguson-Counterpane Internet Security Inc, Doug Kunci Kunci Plaintext Enkripsi Dekripsi Ciphertext Plaintext Jurnal Informatika Mulawarman Vol. 10 No. 1 Februari 2015 23 Whiting-Hi/fn Inc, David Wagner-University of California Berkeley, Chris Hall-Princeton University), Rijndael (Dr. Vincent RijmenKatholieke Universiteit Leuven dan Dr. Joan Daemen-Proton World International), dan RC6 (RSA

Amerika). Setahun kemudian pada tahun 2000, algoritma Rijndael terpilih sebagai algoritma kriptografi yang selain aman juga efisien dalam implementasinya dan dinobatkan sebagai AES. Nama Rijndael sendiri berasal dari gabungan nama penemunya, (Fresly Nandar Pabokory, Indah Fitri Astuti dan Awang Harsa Kridalaksana, 2015:22).

2.1.3 Karakteristik Sistem Kriptografi

Sistem kriptografi dapat di karakteristikan berdasarkan:

1. Tipe operasi dipakai dalam enkripsi dan dekripsi

   Dua tipe yan dipakai dalam enkripsi dan dekripsi subsitusi, elemen pesan (karakter, byte atai bit) ditukar/disubtitusikan dengan elemen lain dariruangan pesan.

2. Tipe kunci yang dipakai

   Umumnya sistem kriptografi kelasik dan beberpa sistem kriptografi modern menggunakan kunci yang sama pada sisi penyadian dan penyulihan sandi. Sistem kriptografi seperti ini disebut dengan kriptografi dengan kunci simetri.

3. Tipe pengolahan pesan

   Ketika melakukan penyandian pesan akan dienkripsi atau dekripsi diolah persatuan blok elemen disebut dengan *Block Chiper*, (Rifky Sadikin, 2012).

2.2 Enkripsi

Ekripsi merupakan hal yang sangat penting dalam kriptografi yang merupakan pengamanan data yan dikirimkan terjaga rahasianya. Pesan asli disebut plaintext yang dirubah menjadi kode-kode yang tidak dimengerti. Enkripsi bisa diartikan dengan chiper atau kode. Sama halnya dengan kita tidak megerti akan sebuah kata, maka kita akan melihatnya di dalam kamus atau daftar istilah-istilah. Beda halnya dengan enkripsi, untuk merubah plaintext ke bentuk chipertext kita menggunakan algoritma yang dapat mengkodekan data yang kita ingini, (Dony Ariyus, 2006).

2.3 Dekripsi

Dekripsi merupakan kebalikan dari enkripsi, pesan yang telah di enkripsi dikembalikan kebentuk asalnya (*Plaintext*) disebut dengan dekripsi pesan. Algoritma yang digunakan untuk dekripsi tentu berbeda dengan yang digunakan untuk enkripsi, (Dony Ariyus, 2006).

2.4 Kunci

Kunci yang dimaksud disini adalah kunci yang dipakai untuk melakukan enkripsi dan dekripsi, kunci terbagi jadi dua bagian kunci pribadi (*private key*) dan kunci umum (*public key*), (Dony Ariyus, 2006).

2.5 Algoritma

Algoritma adalah metode efektif yang diekspresikan sebagai rangkaian terbatas. Algoritma juga merupakan kumpulan perintah untuk menyelesaikan suatu masalah. Perintahperintah ini dapat diterjemahkan secara bertahap dari awal hingga akhir. Masalah tersebut dapat berupa apa saja, dengan syarat untuk setiap permasalahan memiliki kriteria kondisi awal yang harus dipenuhi sebelum menjalankan sebuah algoritma.

Algoritma juga memiliki pengulangan proses (iterasi), dan juga memiliki keputusan hingga keputusan selesai. Dalam cabang disiplin ini, algoritma dipelajari secara abstrak, terlepas dari system komputer atau bahasa pemrograman yang dipergunakan. Algoritma yang berbeda dapat diterapkan untuk suatu permasalahan dengan kriteria yang sama. Kompleksitas dari suatu algoritma merupakan ukuran seberapa banyak komputasi yang diterapkan pada algoritma tersebut untuk menyelesaikan permasalahannya. Secara informal, algoritma yang dapat menyelesaikan permasalahan dalam waktu yang relative singkat memiliki tingkat kompleksitas yang rendah, sementara untuk algoritma yang menyelesaikan permasalahan dalam waktu yang lebih lama memiliki tingkat kompleksitas yang lebih tinggi pula.

Dalam mata kuliah logika dan algoritma, kita telah mempelajari tentang algoritma dan penerapannya dalam pemrograman computer. Kesulitan yang dihadapi dalam permasalahan ini adalah susahnya kita mengerti algoritma dan penyelesaian dari permasalahan yang dihadapi, serta sulitnya membayangkan struktur data yang akan digunakan. Dalam memahami penyelesaian suatu

permasalahan, kita akan lebih mudah untuk mengingat dan memahaminya apabila permasalahan itu dapat ditampilkan dalam bentuk visual dan gambar, sehingga penyajian nya menjadi lebih menarik. Dari permasalahan diatas, penulis ingin membantu mempermudah penyelesaian algoritma untuk mempermudah penyelesaian matematika dengan membuat perangkat lunak alat bantu logika dan algoritma, (Gun Gun Maulan, 2017:69).

Algoritma adalah sistim kerja komputer memiliki brainware, hardware, dan software. Tanpa salah satu dari ketiga sistim tersebut, komputer tidak akan berguna. Kita akan lebih fokus pada softwareckomputer. Software terbangun atas susunan program) dan syntax (cara penulisan/pembuatan program). Untuk menyusun program atau syntax, diperlukannya langkahlangkah yang sistematis dan logis untuk dapat menyelesaikan masalah atau tujuan dalam proses pembuatan suatu software. Maka, algoritma berperan penting dalam penyusunan program atau syntax tersebut. Pengertian algoritma adalah susunan yang logis dan sistematis untuk memecahkan suatu masalah atau untuk mencapai tujuan tertentu. Dalam dunia komputer, algoritma sangat berperan penting dalam pembangunan suatu software. Dalam dunia sehari-hari, mungkin tanpa kita sadari algoritma telah masuk dalam kehidupan kita. Algoritma berbeda dengan logaritma. Logaritma merupakan operasi matematika yang merupakan kebalikan dari eksponen atau pemangkatan. Contoh logaritma seperti bc= a ditulis sebagai blog a = c (b disebut basis), (Gun Gun Maulan, 2017:70).

2.5.1 Struktur Dasar Algoritma

Adapun struktur dasar pada algoritma adalah sebagai berikut:

1. Sekuensial (runtunan) Pada struktur sekuensial ini langkah-langkah yang dilakukan dalam algoritma diproses secara berurutan. Dimulai dari langkah pertama, kedua, dan seterusnya. Pada dasarnya suatu program memang menjalankan suatu proses dari yang dasar seperti struktur ini.
2. Struktur seleksi Struktur seleksi menyatakan pemilihan langkah yang didasarkan oleh suatu kondisi atau pengambilan suatu keputusan. Struktur ini ditandai selalu dengan bentuk flowcart decision (flowcart yang berbentuk belah ketupat). Banyak contoh yang dapat kita terapkan pada struktur jenis ini jika itu menyangkut keputusan, diantaranya: diskon yang berbeda berdasarkan jumlah barang yang ingin dibeli.
3. Struktur perulangan Struktur ini memberikan suatu perintah atau tindakkan yang dilakukan beberapa kali. Misalnya jika teman mau menuliskan kata “belajar c” sebanyak sepuluh kali. Akan lebih efisien jika teman menggunakan sturktur ini dari pada sekedar menuliskannya berturut-turut sebanyak sepuluh kali, (Gun Gun Maulan, 2017 : 70).

2.5.2 Macam-Macam Algoritma Kriptografi

Algoritma kriptografi dibagi menjadi tiga bagian berdasarkan dari kunci yang dipakainya:

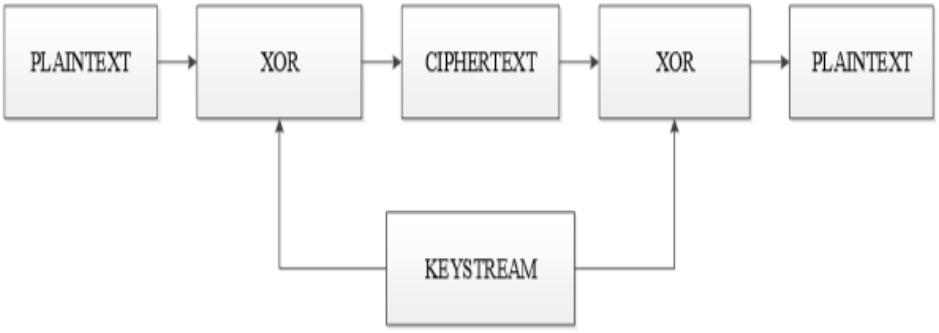
- Algoritma Simetri (menggunakan satu kunci untuk enkripsi dan dekripsinya).
- Algoritma Asimetri (menggunakan kunci yang berbeda untuk enkripsi dan dekripsi).
- Hash Function, (Dony Ariyus, 2006).

2.5.3 Algoritma Vernam *Cipher*

Vernam cipher merupakan algoritma kriptografi yang ditemukan oleh Mayor J. Maugborne dan G. Vernam. Algoritma Vernam *cipher* diadopsi dari *one time pad cipher*, dimana dalam hal ini karakter diganti dengan bit (0 atau 1). Dengan kata lain, vernam *cipher* merupakan versi lain dari *one-time pad cipher*.

Algoritma kriptografi vernam *cipher* merupakan algoritma kriptografi berjenis *symmetric key*. Kunci yang digunakan untuk melakukan enkripsi dan dekripsi menggunakan kunci yang sama. Dalam melakukan proses enkripsi, algoritma vernam *cipher* menggunakan cara stream *cipher* dimana *cipher* berasal dari hasil operasi XOR antara bit plainteks dan bit *key*.

Pada *cipher* aliran, bit hanya mempunyai dua buah nilai, sehingga proses enkripsi hanya menyebabkan dua keadaan pada bit tersebut, yaitu berubah atau tidak berubah. Dua keadaan tersebut ditentukan oleh kunci enkripsi yang disebut dengan aliran-bit-kunci (*keystream*). Secara sederhana proses enkripsi dan dekripsi algoritma vernam *cipher* dapat adalah pada gambar dibawah ini: (Mohammad Jumeidi, Dedi Triyanto dan Yulrio Brianorman, 2016:22)



Gambar 2.2 Proses Enkripsi dan Dekripsi Algoritma Kriptografi Vernam Cipher

Satu *vernam cipher* menghasilkan apa yang disebut suatu *keystream* (suatu barisan bit yang digunakan sebagai kunci). Proses enkripsi dicapai dengan menggabungkan *keystream* dengan *plaintext* biasanya dengan operasi bitwise XOR. Pembentukan *keystream* dapat dibuat independen terhadap *plaintext* dan *ciphertext*, menghasilkan *synchronous stream cipher*, atau dapat dibuat tergantung pada data dan enkripsinya, dalam hal mana stream *cipher* disebut sebagai *self-synchronizing*. Kebanyakan bentuk stream *cipher* adalah *synchronous stream cipher*.

Konsentrasi dalam *stream ciphers* pada umumnya berkaitan dengan sifat sifat teoritis yang menarik dari onetime pad. Suatu one-time pad, kadang-kadang disebut *Vernam cipher*, menggunakan sebuah string dari bit yang dihasilkan murni secara random. Keystream memiliki panjang sama dengan pesan plaintext; string random digabungkan dengan menggunakan *bitwise* XOR dengan *plaintext* untuk menghasilkan *ciphertext*. Karena *keystream* seluruhnya adalah random, walaupun dengan sumber daya komputasi tak terbatas seseorang hanya dapat menduga *plaintext* jika melihat *ciphertext*. Metode *cipher* seperti ini disebut memberikan kerahasiaan yang sempurna (*perfect secrecy*). Metode *vernam cipher* yang umum digunakan adalah RC4. Satu hal yang menarik bahwa mode operasi tertentu dari suatu *block cipher* dapat mentransformasikan secara efektif hasil operasi tersebut ke dalam satu keystream generator dan dalam hal ini, *block cipher* apa saja dapat digunakan sebagai suatu stream cipher; seperti dalam DES, CFB atau OFB. Akan tetapi, *vernam ciphers* dengan desain khusus biasanya jauh lebih cepat. (Eko Hari Rachmawanto, Christy Atika Sari, Yani Parti Astuti dan Liya Umaroh, 2016:47)

2.5.4 Algoritma Kriptografi

Algoritma ditinjau dari asal usul kata, kata algoritma mempunyai sejarah yang menarik, kata ini muncul di dalam kamus Webster sampai akhir tahun 1957 hanya menemukan kata algorism yang mempunyai arti proses perhitungan dengan bahasa Arab. Algoritma berasal dari nama penulis buku Arab yang terkenal yaitu Abu Ja far Muhammad ibnu Musa al-Khuwarizmi (al-Khuwarizmi dibaca oleh orang barat menjadi algorism). Kata algorism lambat laun berubah menjadi

algorithm. Definisi terminologinya Algoritma adalah urutan langkah-langkah logis untuk penyelesaian masalah yang disusun secara sistimatis. Algoritma kriptografi merupakan langkah-langkah logis bagaimana menyembunyi- can pesan dari orang-orang yang tidak berhak atas pesan tersebut, (Dony Ariyus, 2016)

2.6 Ascii Code

Berikut ini adalah ascii code yang berguna untuk penyandian pesan:

		Simbol	
ascii code	0	NULL	(Null character)
ascii code	1	SOH	(Start of Header)
ascii code	2	STX	(Start of Text)
ascii code	3	ETX	(End of Text)
ascii code	4	EOT	(End of Transmission)
ascii code	5	ENQ	(Enquiry)
ascii code	6	ACK	(Acknowledgement)
ascii code	7	BEL	(Bell)
ascii code	8	BS	(Backspace)
ascii code	9	HT	(Horizontal Tab)
ascii code	10	LF	(Line feed)
ascii code	11	VT	(Vertical Tab)
ascii code	12	FF	(Form feed)
ascii code	13	CR	(Carriage return)
ascii code	14	SO	(Shift Out)
ascii code	15	SI	(Shift In)
ascii code	16	DLE	(Data link escape)
ascii code	17	DC1	(Device control 1)
ascii code	18	DC2	(Device control 2)
ascii code	19	DC3	(Device control 3)
ascii code	20	DC4	(Device control 4)
ascii code	21	NAK	(Negative acknowledgement)
ascii code	22	SYN	(Synchronous idle)
ascii code	23	ETB	(End of transmission block)
ascii code	24	CAN	(Cancel)
ascii code	25	EM	(End of medium)

ascii code	26	SUB	(Substitute)
ascii code	27	ESC	(Escape)
ascii code	28	FS	(File separator)
ascii code	29	GS	(Group separator)
ascii code	30	RS	(Record separator)
ascii code	31	US	(Unit separator)
ascii code	32		(Space)
ascii code	33	!	(Exclamation mark)
ascii code	34	"	(Quotation mark ; quotes)
ascii code	35	#	(Number sign)
ascii code	36	$	(Dollar sign)
ascii code	37	%	(Percent sign)
ascii code	38	&	(Ampersand)
ascii code	39	'	(Apostrophe)
ascii code	40	(	(round brackets or parentheses)
ascii code	41	)	(round brackets or parentheses)
ascii code	42	*	(Asterisk)
ascii code	43	+	(Plus sign)
ascii code	44	,	(Comma)
ascii code	45	-	(Hyphen)
ascii code	46	.	(Dot , full stop)
ascii code	47	/	(Slash)
ascii code	48	0	(number zero)
ascii code	49	1	(number one)
ascii code	50	2	(number two)
ascii code	51	3	(number three)
ascii code	52	4	(number four)
ascii code	53	5	(number five)
ascii code	54	6	(number six)
ascii code	55	7	(number seven)
ascii code	56	8	(number eight)
ascii code	57	9	(number nine)
ascii code	58	:	(Colon)
ascii code	59	;	(Semicolon)
ascii code	60	<	(Less-than sign)
ascii code	61	=	(Equals sign)
ascii code	62	>	(Greater-than sign ; Inequality)
ascii code	63	?	(Question mark)
ascii code	64	@	(At sign)

ascii code	65	A	(Capital A)
ascii code	66	B	(Capital B)
ascii code	67	C	(Capital C)
ascii code	68	D	(Capital D)
ascii code	69	E	(Capital E)
ascii code	70	F	(Capital F)
ascii code	71	G	(Capital G)
ascii code	72	H	(Capital H)
ascii code	73	I	(Capital I)
ascii code	74	J	(Capital J)
ascii code	75	K	(Capital K)
ascii code	76	L	(Capital L)
ascii code	77	M	(Capital M)
ascii code	78	N	(Capital N)
ascii code	79	O	(Capital O)
ascii code	80	P	(Capital P)
ascii code	81	Q	(Capital Q)
ascii code	82	R	(Capital R)
ascii code	83	S	(Capital S)
ascii code	84	T	(Capital T)
ascii code	85	U	(Capital U)
ascii code	86	V	(Capital V)
ascii code	87	W	(Capital W)
ascii code	88	X	(Capital X)
ascii code	89	Y	(Capital Y)
ascii code	90	Z	(Capital Z)
ascii code	91	[	(square brackets or box brackets)
ascii code	92	\	(Backslash)
ascii code	93	]	(square brackets or box brackets)
ascii code	94	^	(Caret or circumflex accent)
ascii code	95	_	(underscore , understrike , underbar or low line)
ascii code	96	`	(Grave accent)
ascii code	97	a	(Lowercase a)
ascii code	98	b	(Lowercase b)
ascii code	99	c	(Lowercase c)
ascii code	100	d	(Lowercase d)
ascii code	101	e	(Lowercase e)
ascii code	102	f	(Lowercase f)

ascii code	103	g	(Lowercase g)
ascii code	104	h	(Lowercase h)
ascii code	105	i	(Lowercase i)
ascii code	106	j	(Lowercase j)
ascii code	107	k	(Lowercase k)
ascii code	108	l	(Lowercase l)
ascii code	109	m	(Lowercase m)
ascii code	110	n	(Lowercase n)
ascii code	111	o	(Lowercase o)
ascii code	112	p	(Lowercase p)
ascii code	113	q	(Lowercase q)
ascii code	114	r	(Lowercase r)
ascii code	115	s	(Lowercase s)
ascii code	116	t	(Lowercase t)
ascii code	117	u	(Lowercase u)
ascii code	118	v	(Lowercase v)
ascii code	119	w	(Lowercase w)
ascii code	120	x	(Lowercase x)
ascii code	121	y	(Lowercase y)
ascii code	122	z	(Lowercase z)
ascii code	123	{	(curly brackets or braces)
ascii code	124	|	(vertical-bar, vbar, vertical line or vertical slash)
ascii code	125	}	(curly brackets or braces)
ascii code	126	~	(Tilde ; swung dash)
ascii code	127	DEL	(Delete)
ascii code	128	Ç	(Majuscule C-cedilla)
ascii code	129	ü	(letter "u" with umlaut or diaeresis ; "u-umlaut")
ascii code	130	é	(letter "e" with acute accent or "e-acute")
ascii code	131	â	(letter "a" with circumflex accent or "a-circumflex")
ascii code	132	ä	(letter "a" with umlaut or diaeresis ; "a-umlaut")
ascii code	133	à	(letter "a" with grave accent)
ascii code	134	å	(letter "a" with a ring)
ascii code	135	ç	(Minuscule c-cedilla)
ascii code	136	ê	(letter "e" with circumflex accent or "e-circumflex")
ascii code	137	ë	(letter "e" with umlaut or diaeresis ; "e-umlaut")

ascii code	138	è	(letter "e" with grave accent)
ascii code	139	ï	(letter "i" with umlaut or diaeresis ; "i-umlaut")
ascii code	140	î	(letter "i" with circumflex accent or "i-circumflex")
ascii code	141	ì	(letter "i" with grave accent)
ascii code	142	Ä	(letter "A" with umlaut or diaeresis ; "A-umlaut")
ascii code	143	Å	(Capital letter "A" with a ring)
ascii code	144	É	(Capital letter "E" with acute accent or "E-acute")
ascii code	145	æ	(Latin diphthong "ae" in lowercase)
ascii code	146	Æ	(Latin diphthong "AE" in uppercase)
ascii code	147	ô	(letter "o" with circumflex accent or "o-circumflex")
ascii code	148	ö	(letter "o" with umlaut or diaeresis ; "o-umlaut")
ascii code	149	ò	(letter "o" with grave accent)
ascii code	150	û	(letter "u" with circumflex accent or "u-circumflex")
ascii code	151	ù	(letter "u" with grave accent)
ascii code	152	ÿ	(Lowercase letter "y" with diaeresis)
ascii code	153	Ö	(letter "O" with umlaut or diaeresis ; "O-umlaut")
ascii code	154	Ü	(letter "U" with umlaut or diaeresis ; "U-umlaut")
ascii code	155	ø	(slashed zero or empty set)
ascii code	156	£	(Pound sign ; symbol for the pound sterling)
ascii code	157	Ø	(slashed zero or empty set)
ascii code	158	×	(multiplication sign)
ascii code	159	*ƒ*	(function sign ; f with hook sign ; florin sign)
ascii code	160	á	(letter "a" with acute accent or "a-acute")
ascii code	161	í	(letter "i" with acute accent or "i-acute")
ascii code	162	ó	(letter "o" with acute accent or "o-acute")
ascii code	163	ú	(letter "u" with acute accent or "u-acute")
ascii code	164	ñ	(letter "n" with tilde ; enye)
ascii code	165	Ñ	(letter "N" with tilde ; enye)
ascii code	166	ª	(feminine ordinal indicator)
ascii code	167	º	(masculine ordinal indicator)
ascii code	168	¿	(Inverted question marks)

ascii code	169	®	(Registered trademark symbol)
ascii code	170	¬	(Logical negation symbol)
ascii code	171	½	(One half)
ascii code	172	¼	(Quarter or one fourth)
ascii code	173	¡	(Inverted exclamation marks)
ascii code	174	«	(Angle quotes or guillemets)
ascii code	175	»	(Guillemets or angle quotes)
ascii code	176	░	
ascii code	177	▒	
ascii code	178	▓	
ascii code	179	│	(Box drawing character)
ascii code	180	┤	(Box drawing character)
ascii code	181	Á	(Capital letter "A" with acute accent or "A-acute")
ascii code	182	Â	(letter "A" with circumflex accent or "A-circumflex")
ascii code	183	À	(letter "A" with grave accent)
ascii code	184	©	(Copyright symbol)
ascii code	185	╣	(Box drawing character)
ascii code	186	║	(Box drawing character)
ascii code	187	╗	(Box drawing character)
ascii code	188	╝	(Box drawing character)
ascii code	189	¢	(Cent symbol)
ascii code	190	¥	(YEN and YUAN sign)
ascii code	191	┐	(Box drawing character)
ascii code	192	└	(Box drawing character)
ascii code	193	┴	(Box drawing character)
ascii code	194	┬	(Box drawing character)
ascii code	195	├	(Box drawing character)
ascii code	196	─	(Box drawing character)
ascii code	197	┼	(Box drawing character)
ascii code	198	ã	(Lowercase letter "a" with tilde or "a-tilde")
ascii code	199	Ã	(Capital letter "A" with tilde or "A-tilde")
ascii code	200	╚	(Box drawing character)
ascii code	201	╔	(Box drawing character)
ascii code	202	╩	(Box drawing character)
ascii code	203	╦	(Box drawing character)
ascii code	204	╠	(Box drawing character)
ascii code	205	═	(Box drawing character)
ascii code	206	╬	(Box drawing character)

ascii code	207	¤	(generic currency sign)
ascii code	208	ð	(Lowercase letter "eth")
ascii code	209	Ð	(Capital letter "Eth")
ascii code	210	Ê	(letter "E" with circumflex accent or "E-circumflex")
ascii code	211	Ë	(letter "E" with umlaut or diaeresis ; "E-umlaut")
ascii code	212	È	(letter "E" with grave accent)
ascii code	213	ı	(lowercase dot less i)
ascii code	214	Í	(Capital letter "I" with acute accent or "I-acute")
ascii code	215	Î	(letter "I" with circumflex accent or "I-circumflex")
ascii code	216	Ï	(letter "I" with umlaut or diaeresis ; "I-umlaut")
ascii code	217	┘	(Box drawing character)
ascii code	218	┌	(Box drawing character)
ascii code	219	█	(Block)
ascii code	220	▄	
ascii code	221	¦	(vertical broken bar)
ascii code	222	Ì	(letter "I" with grave accent)
ascii code	223	▀	
ascii code	224	Ó	(Capital letter "O" with acute accent or "O-acute")
ascii code	225	ß	(letter "Eszett" ; "scharfes S" or "sharp S")
ascii code	226	Ô	(letter "O" with circumflex accent or "O-circumflex")
ascii code	227	Ò	(letter "O" with grave accent)
ascii code	228	õ	(letter "o" with tilde or "o-tilde")
ascii code	229	Õ	(letter "O" with tilde or "O-tilde")
ascii code	230	µ	(Lowercase letter "Mu" ; micro sign or micron)
ascii code	231	þ	(Lowercase letter "Thorn")
ascii code	232	Þ	(Capital letter "thorn")
ascii code	233	Ú	(Capital letter "U" with acute accent or "U-acute")
ascii code	234	Û	(letter "U" with circumflex accent or "U-circumflex")
ascii code	235	Ù	(letter "U" with grave accent)
ascii code	236	ý	(Lowercase letter "y" with acute accent)
ascii code	237	Ý	(Capital letter "Y" with acute accent)
ascii code	238	¯	(macron symbol)

ascii code	239	´	(Acute accent)
ascii code	240	-	(Hyphen)
ascii code	241	±	(Plus-minus sign)
ascii code	242	‗	(underline or underscore)
ascii code	243	¾	(three quarters)
ascii code	244	¶	(paragraph sign or pilcrow)
ascii code	245	§	(Section sign)
ascii code	246	÷	(The division sign ; Obelus)
ascii code	247	¸	(cedilla)
ascii code	248	°	(degree symbol)
ascii code	249	¨	(Diaeresis)
ascii code	250	·	(Interpunct or space dot)
ascii code	251	¹	(superscript one)
ascii code	252	³	(cube or superscript three)
ascii code	253	²	(Square or superscript two)
ascii code	254	■	(black square)
ascii code	255	nbsp	(non-breaking space or no-break space)

Tabel 2.1 ascii code

2.7 Biner

Biner merupakan suatu penulisan angka yaitu 0 dan 1, digunakan untuk suatu perhitungan atau penyandian suatu pesan.

Desimal	Biner (8 bit)
0	0000 0000
1	0000 0001
2	0000 0010
3	0000 0011
4	0000 0100
5	0000 0101

6	0000 0110
7	0000 0111
8	0000 1000
9	0000 1001
10	0000 1010
11	0000 1011
12	0000 1100
13	0000 1101
14	0000 1110
15	0000 1111
16	0001 0000
17	0001 0001
18	0001 0010
19	0001 0011
20	0001 0100
21	0001 0101
22	0001 0110
23	0001 0111
24	0001 1000
25	0001 1001
26	0001 1010
27	0001 1011

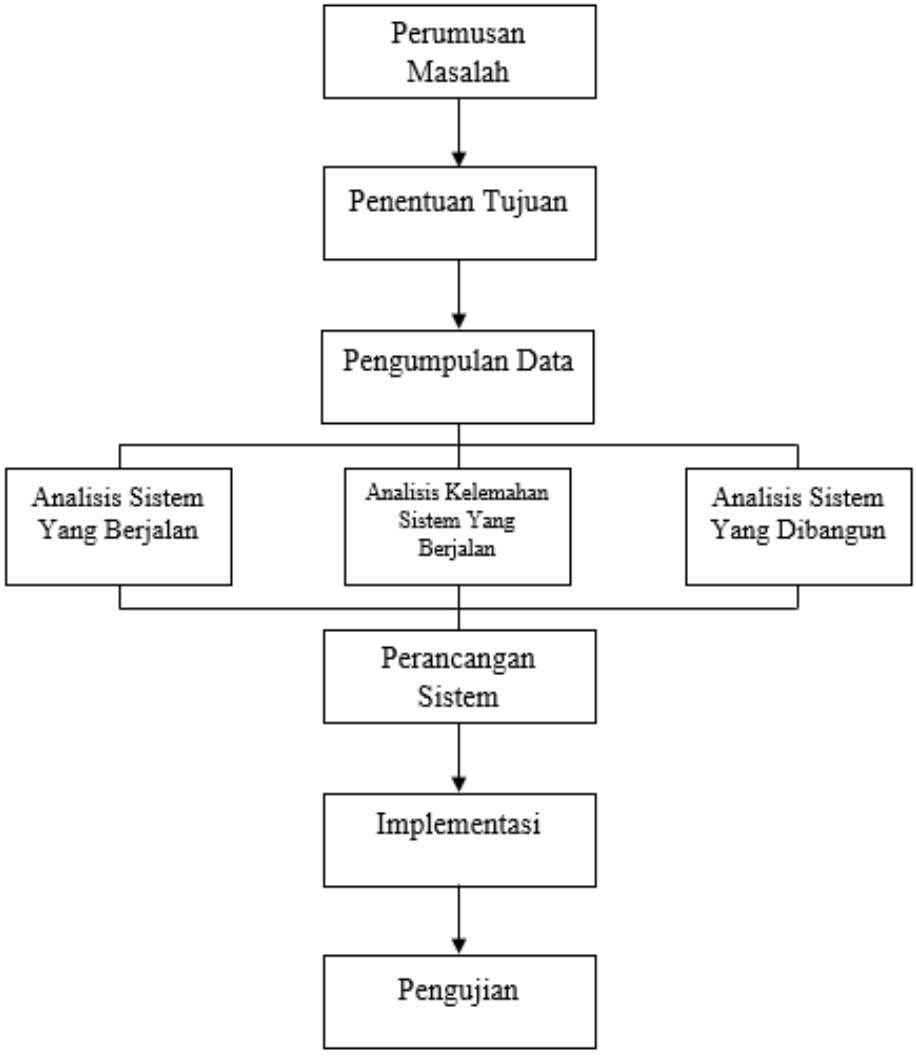
28	0001 1100
29	0001 1101
30	0001 1110

Tabel 2.2 Bilangan Biner

BAB III

METODOLOGI PENELITIAN

3.1 Tahapan Penelitian



Gambar 3.1 Tahapan penelitian

3.2 Metode Pengumpulan Data

Metode pengumpulan data dapat didefinisikan sebagai salah satu cara yang digunakan untuk memperoleh data yang dibutuhkan. Sebagai bahan masukan bagi peneliti dalam penyusunan skripsi ini adalah sebagai berikut:

1. Studi literature

   Dalam melakukam penelitian ilmiah harus dilakukan teknik penyusunan yang sistematis untuk memudahkan langkah – langkah yang akan diambil. Begitu pula yang dilakukan penulis dalam penelitian ini, yang harus dilakukan yaitu dengan melakukan studi literatur pada buku – buku yang membahas tentang *Vernam Chiper.*

2. Studi Pustaka

   Penelitian ini bersifat teoritis dengan cara memperoleh informasi dalam buku bacaan yang berhubungan dengan masalah yang akan dibahas. Dalam menyelesaikan skripsi ini, peneliti membutuhkan data yang berhubungan dengan Sistem Informasi ini.

3.3 Analisis Sistem Sedang Berjalan

Pada perancangan aplikasi Enkripsi dan Dekripsi dengan teknik XOR dengan menggunakan metode vernam cipher untuk mendapatkan hasil teks yang diubah (*Chipertext)*, Menggunakan angka dan tabel untuk konversi dibutuhkan sebuah perhitungan yang sesuai dengan metode yang digunakan.

Adapun rumus dan perhitungan metode yang digunakan adalah sebagai berikut :

XOR

0	0	0
0	1	1
1	0	1
1	1	0

XNOR

0	0	1
0	1	0
1	0	0
1	1	1

Plaintext "**HALO APA KABAR**"

Kunci "**SAYA"**

Blok Kunci "**SAYASAYASAYASA"**

Maka untuk mendapatkan *Ciphertext*nya harus menggunakan penghitungan seperti di bawah ini:

Langkah Pertama membuat tabel konversi ASCII.

Ciphertext : HALO APA KABAR

Kunci : SAYA

Penerima memilih kata SAYA sebagai kunci yang akan digunakan untuk melakukan proses enkripsi menggunakan Algoritma *Vernam Cipher*, sehingga pada prosesnya kata SAYA akan mengikuti banyak karakter *Ciphertext* 1 yang didapat.

Ciphertext : HALO APA KABAR

Blok Kunci : SAYASAYASAYASA

3.3.1 Kelemahan Sistem Yang Sedang Berjalan

Setelah menganalisis sistem yang sedang berjalan penulis dapat menguraikan beberapa kelemahan – kelemahan dari proses enkripsi dan dekripsi

data diantaranya :

a. Diharuskan bagi penulis untuk melihat tabel asci untuk proses penyandian teks
b. Jika tulisan terlalu banyak, menambah kesulitan pada proses penyandian.
c. Memungkinkan kesalahan pada proses penyandian

3.3.2 Analisis Sistem yang Dibangun

Perancangan sistem yang akan dibangun dilakukan setelah menganalisis permasalahan yang ada dari sistem berjalan. Sistem baru yang akan dibangun ini merupakan perubahan dari sistem yang dilakukan secara manual yang akan dijadikan secara komputerisasi dengan menggunakan aplikasi visual studio.

3.4 Metode Perancangan Penelitian

Perancangan sistem dilaksanakan dengan menyesuaikan konseptual desain yang kemudian diterapkan pada tahap penulisan kode dan perancangan aplikasi. Bagian dari perancangan sistem ini berisi penjelasan mengenai analisa dan perancangan aplikasi enkripsi dan dekripsi dengan teknik xor dengan menggunakan *Vernam Cipher* meliputi :

1. Desain Sistem

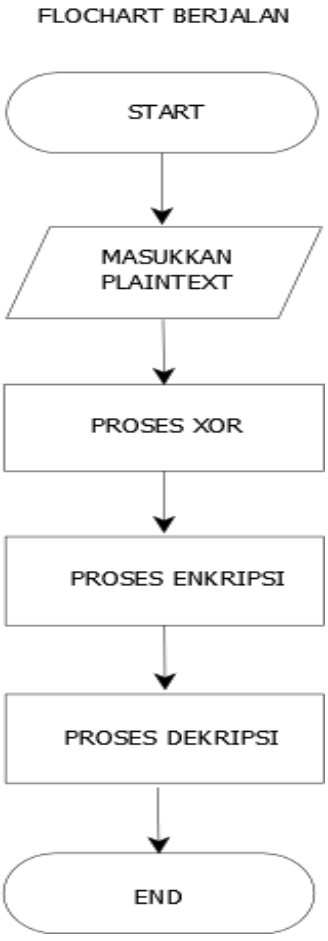
   Desain sistem digunakan peneliti sebagai penggambaran perencanaan dan pembuatan sketsa pada sistem yang akan dibangun.

2. Perancangan Sistem

Merancang alir kerja *(workflow)* dari sistem dalam bentuk *diagaram* alir (*flowchart)* atau UML.

3.4.1 *Flowchart* Berjalan

Adapun *flowchart* berjalan dari sistem aplikasi enkripsi dan dekripsi dengan menggunakan teknik XOR menggunakan metode *Vernam Chiper* dapat dirincikan seperti pada gambar 3.2:

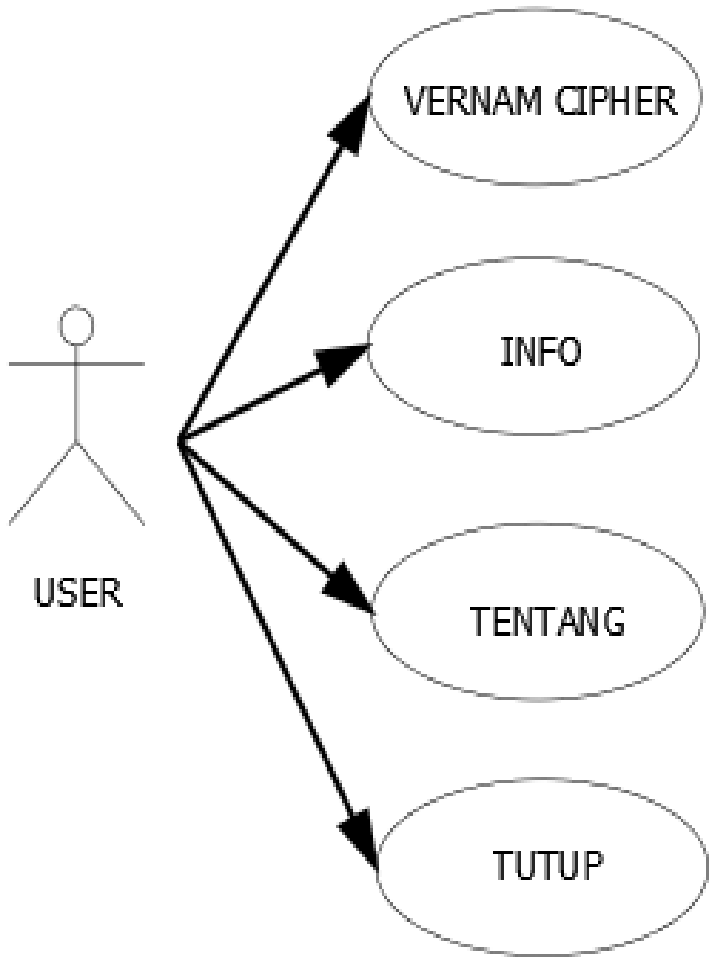


Gambar 3.2 *Flochart* Berjalan

Gambar diatas menjelaskan bahwa ketika user memulai sebuah sistem, user akan memasukkan sebuah kata atau kalimat yang disebut plaintext kemudian melakukan perhitungan dengan teknik xor , maka dapat ditampilan hasil enkripsi dan dekripsi.

3.4.2 *Use Case*

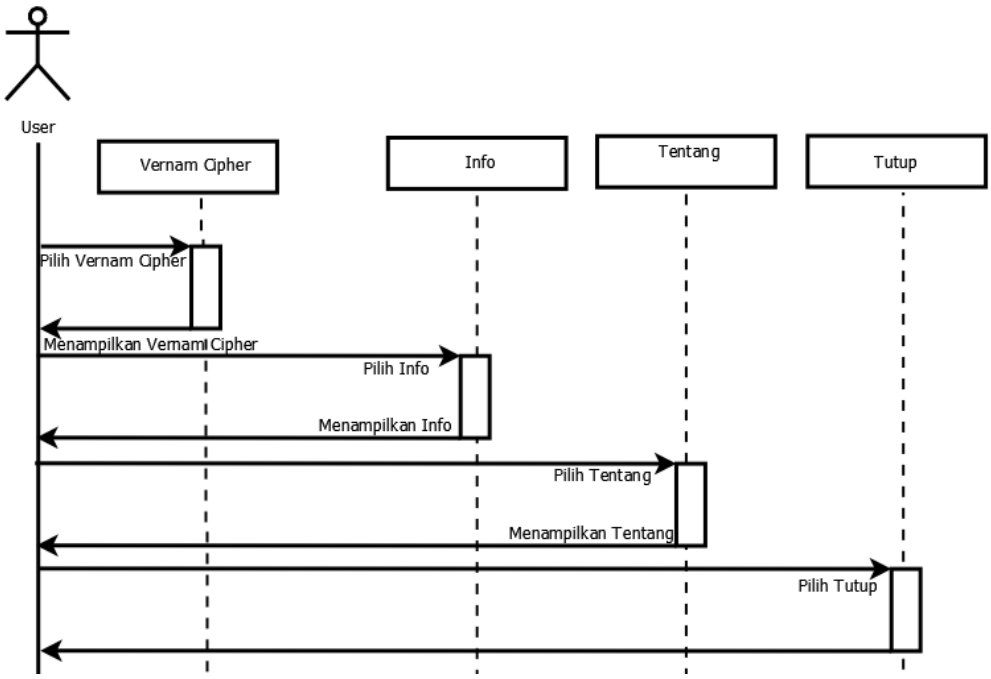
Adapun *use case* dari sistem aplikasi enkripsi dan dekripsi dengan menggunakan teknik XOR menggunakan metode *Vernam Chiper* dapat dirincikan seperti pada gambar 3.3:



Gambar 3.3 *Use Case* Berjalan

a. *Use Case Vernam Cipher* berfungsi untuk menampilkan aplikasi dimana didalamnya terdapat plaintext , kunci , dan chipertext agar dapat mempermudah pengguna dalam melakukan enkripsi dan dekripsi data sehingga terjamin keamanannya.
b. *Use Case* Info berfungsi untuk menampilkan penjelasan tentang aplikasi yang dibangun dan menjelaskan enkripsi dan dekripsi data.
c. *Use Case* Tentang berfungsi untuk menampilkan profil penulis.
d. *Use Case* Tutup berfungsi untuk keluar dari aplikasi enkripsi dan dekripsi sehingga langsung diarahkan ke Menu Utama.

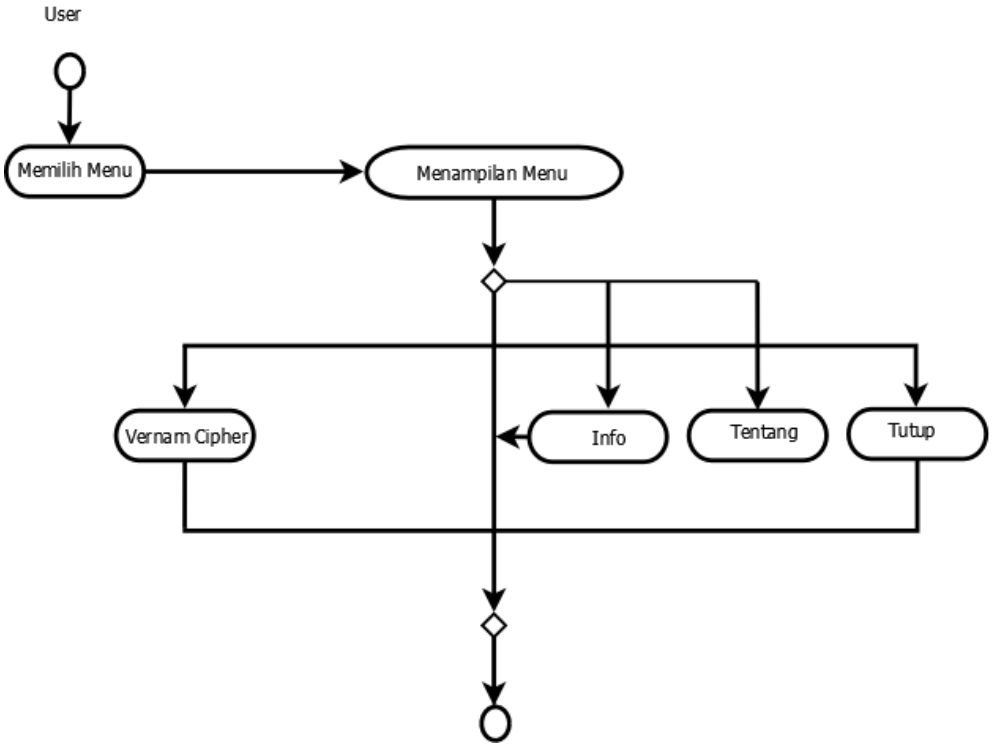
3.4.3 *Sequence Diagram*



Gambar 3.4 *Sequence Diagram*

3.4.4 *Activity Diagram*

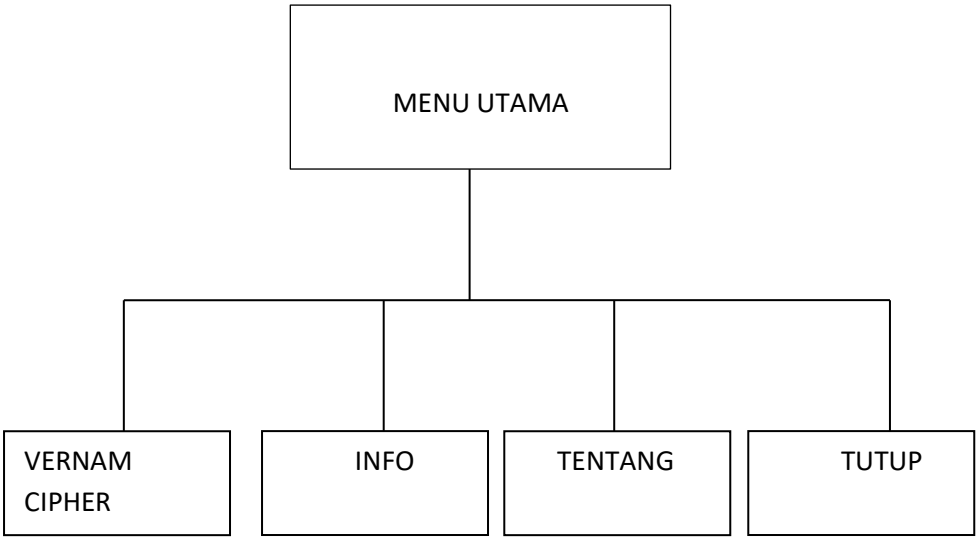
Activity diagram menggambarkan aktifitas - aktifitas yang terjadi dalam aplikasi dari aktivitas dimulai sampai aktivitas berhenti.



Gambar 3.5 *Activity Diagram*

3.5 Rancangan Tampilan

Berikut ini adalah rancangan tampilan yang akan peneliti bangun, yakni aplikasi enkripsi dan dekripsi dengan teknik xor menggunakan metode *vernam cipher.*

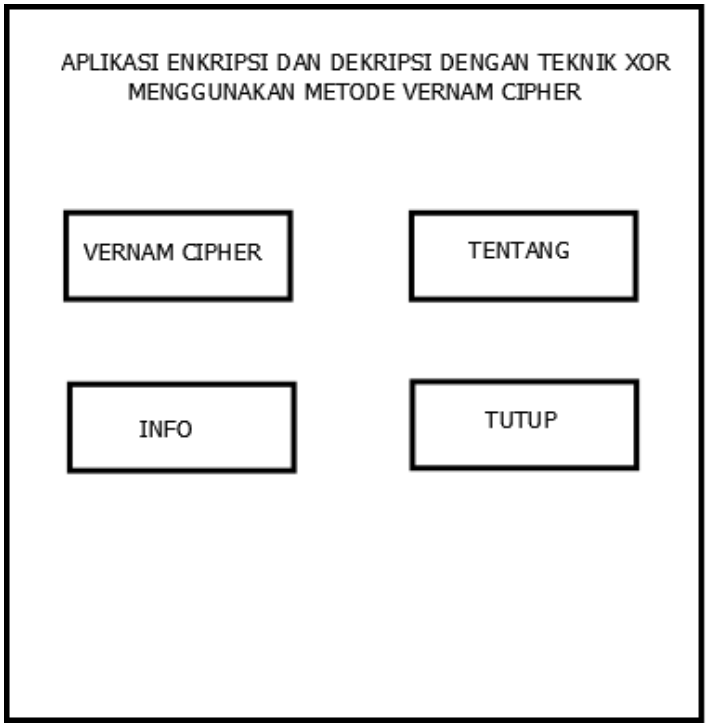


Gambar 3.6 Rancangan Tampilan

3.6 Perancangan Antarmuka

1. Halaman Menu Utama

   Form ini berisi tombol-tombol seperti menu *Vernam Cipher*, Info, Tentang, dan Tutup.



Gambar 3.7 Rancangan Halaman Menu Utama

Pada gambar di atas terdapat 4 menu yaitu Vernam Cipher, Info, Tentang dan Tutup.

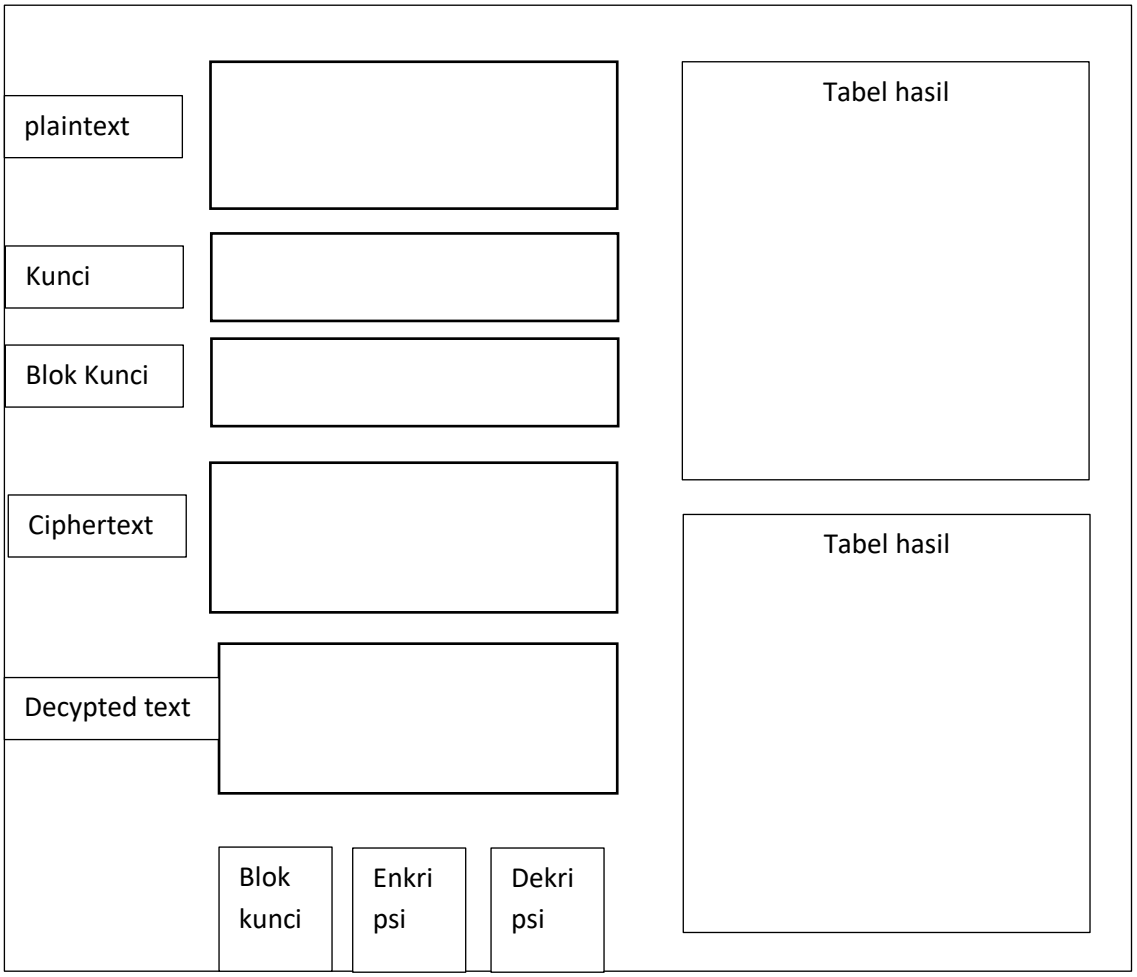
Menu Materi berfungsi untuk menghubungkan pengguna ke form materi.

a. Menu Vernam Cipher berfungsi untuk menghubungkan pengguna ke aplikasi enkripsi dan dekripsi.
b. Menu Info berfungsi untuk menghubungkan pengguna ke form tentang.
c. Menu Tutup berfungsi untuk keluar dari program.

2. Halaman Vernam Cipher

   Form ini berisi penjelasan mengenai Vernam Cipher. Dimana pengguna memasukkan tulisan asli atau *plaintext* ke dalam kolom *plaintext* kemudian masukkan kunci beserta blok kunci, setelah itu pilih tombol

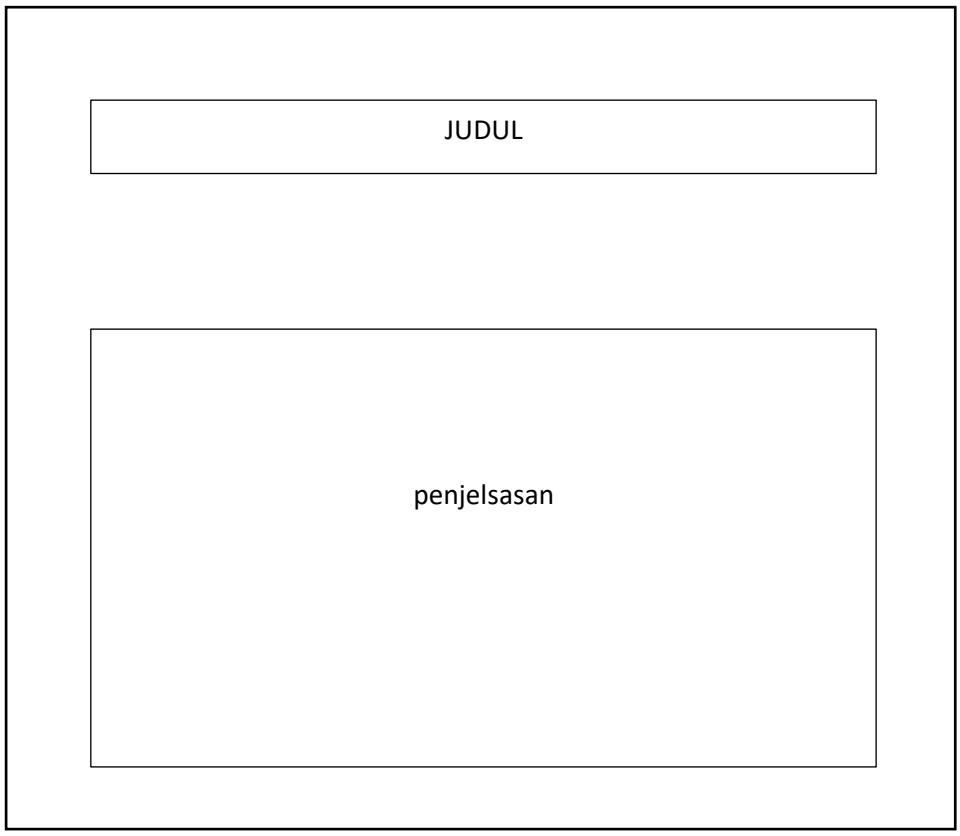
   Enkripsi untuk mengacak pesan dengan bantuan kunci yang telah dibuat sebelumnya, dan tekan tombol Dekripsi untuk mengembalikan isi pesan kebentuk awal.



Gambar 3.8 Halaman Vernam Cipher

3. Tampilan Menu Info

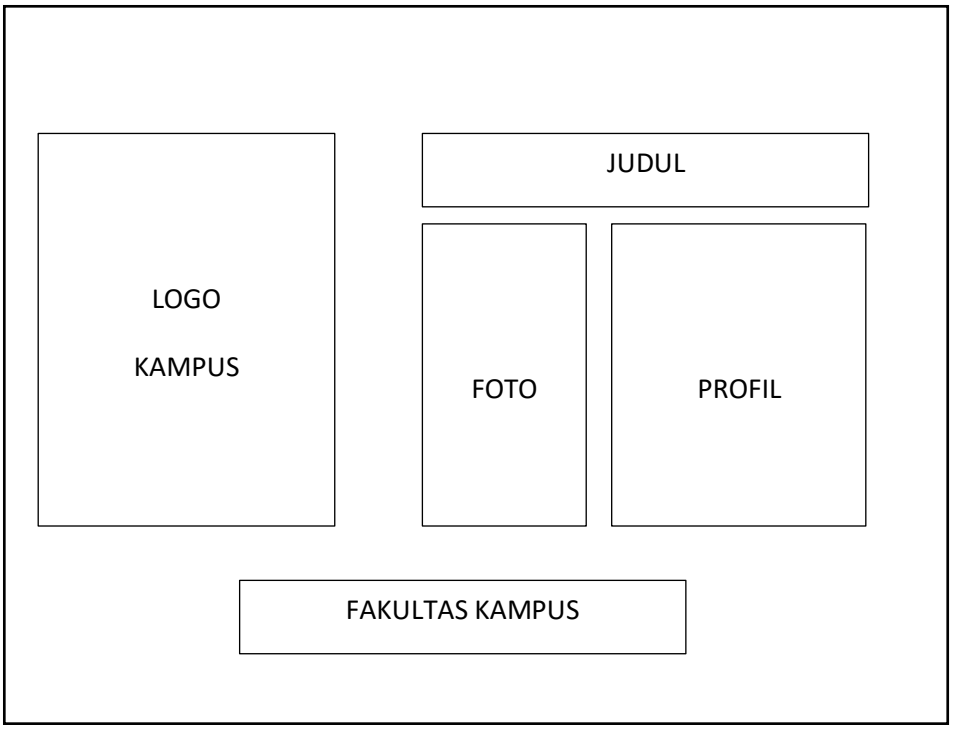
Form ini digunakan untuk menjelaskan proses pengamanan suatu informasi agar tidak dapat dibaca tanpa bantuan khusus dan menjelaskan cara kerja penyandian, dimulai dari plaintext kemudian kunci yang dikonversikan dalam bentuk angka dan dibalikkan ke huruf.



Gambar 3.9 Tampilan Menu Info

4. Tampilan Menu Tentang

   Form ini berisi tentang info profil dari penulis.



Gambar 3.10 Tampilan Menu Tentang

BAB IV

HASIL DAN PEMBAHASAN

4.1 Kebutuhan Spesifikasi *Hardware* dan *Software*

Untuk dapat menjalankan sistem aplikasi enkripsi dan dekripsi dengan teknik XOR menggunakan metode Vernam Cipher perlunya spesifikasi Hardware dan Software yang mendukung adalah sebagai berikut :

4.1.1 Antar Muka Hardware

Antarmuka sistem dengan perangkat keras (*hardware)* di mana sistem berinteraksi harus didefinisikan dengan detail. Antarmuka *hardware* menerangkan karakteristik logika, protokol yang dipakai dan modus operasi. *Hardware* yang digunakan sebagai pengoperasian adalah komputer dengan spesifikasi seperti :

a. Laptop 14”
b. Processor Intel® Core ™ i3-4030U CPU 1.90 GHz
c. RAM 4 GB
d. Keyboard dan Mouse

4.1.2 Antar Muka Software

Untuk menjalankan sistem, dibutuhkan spesifikasi minimal perangkat lunak pendukung utama dalam *computer*. Adapun perangkat lunak tersebut seperti:

a. Microsoft Windows 10
b. Visual Studio

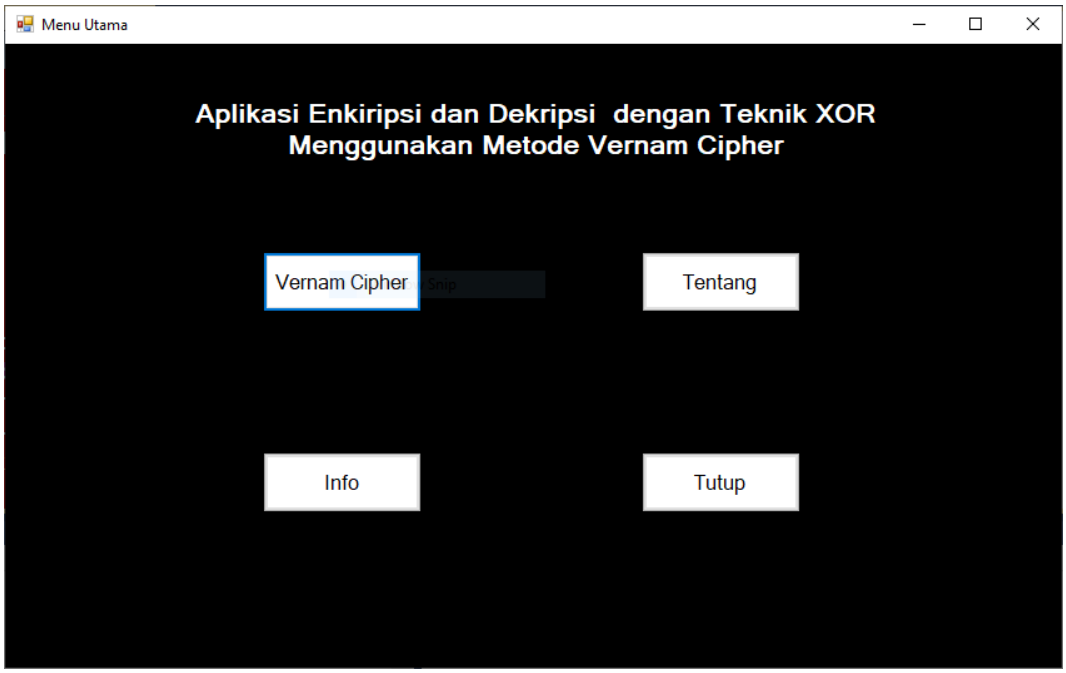
4.2 Perancangan

Pengujian merupakan bagian yang penting dalam siklus pengembangan perangkat lunak. Pengujian dilakukan untuk menjamin kualitas dan juga mengetahui kelemahan dari perangkat lunak. Tujuan dari pengujian ini adalah untuk menjamin bahwa perangkat lunak yang dibangun memiliki kualitas yang handal. Pengujian terhadap program itu sendiri yang bertujuan agar program dapat berjalan dengan baik tanpa mengalami gangguan atau *error*, dan memungkinkan untuk dilakukannya pengembangan sistem lebih lanjut. Pengujian perangkat lunak ini menggunakan metode pengujian *black box.* Pengujian *black box* ini tidak perlu tahu apa yang sesungguhnya terjadi dalam sistem atau perangkat lunak, yang diuji adalah masukan serta keluarannya. Berikut ini adalah rencana pengujian input/output aplikasi enkripsi dan dekripsi.

4.3 Implementasi

Implementasi adalah proses penerapan rancangan program yang telah dibuat pada bab sebelumnya dalam melaksanakan sistem informasi pemrograman yang telah dibuat, hasil dari tahapan implementasi ini adalah suatu sistem pengolahan data yang sudah dapat berjalan dengan baik. Dengan demikian dapat diketahui apakah perangkat lunak ini dapat menghasilkan sistem aplikasi enkripsi dan dekripsi terbaik yang sesuai dengan tujuan yang diharapkan.

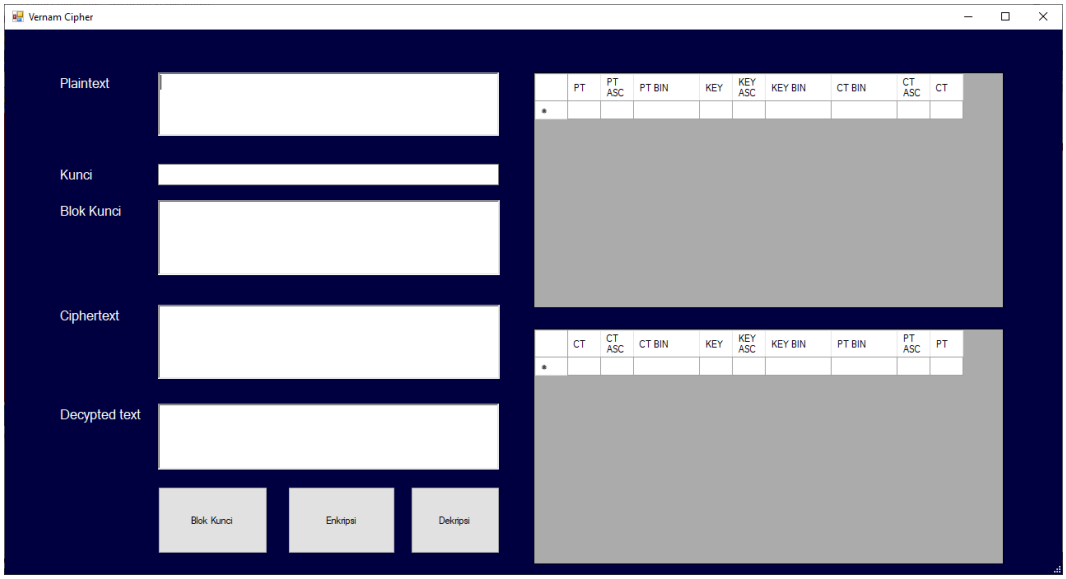
4.3.1 Tampilan Menu Utama



Gambar 4.1 Tampilan Menu Utama

Gambar diatas merupakan tampilan menu utama pada aplikasi enkripsi dan dekripsi untuk menampilkan vernam cipher, info, tentang dan menutup aplikasi.

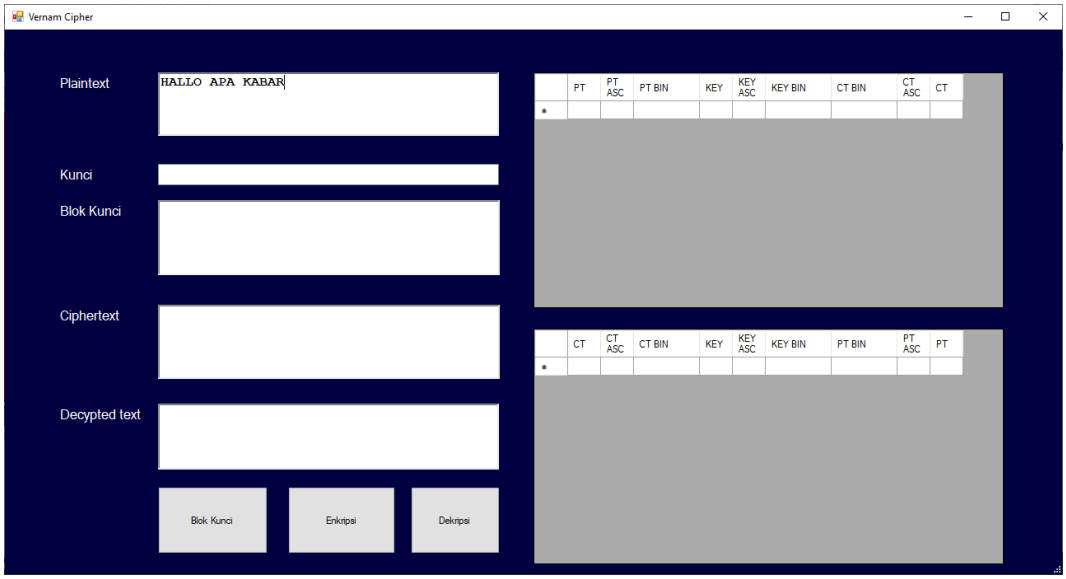
4.3.2 Tampilan awal Vernam Cipher



Gambar 4.2 Tampilan awal Vernam Cipher

Gambar di atas menunjukan tampilan awal Vernam Cipher dimana pada gambar di atas menunjukan plaintext untuk memasukan pesan, kunci untuk memasukan kuci, blok kunci untuk membuat kunci menjadi diblok atau diulang, ciphertext untuk menampilkan hasil dari enkripsi dan decypted text itu untuk menampilkan hasil dari dekripsi. Di bagian kanan terdapat dua tabel yaitu tabel yang di atas adalah tabel enkripsi dan tabel yang dibawah adalah tabel dekripsi.

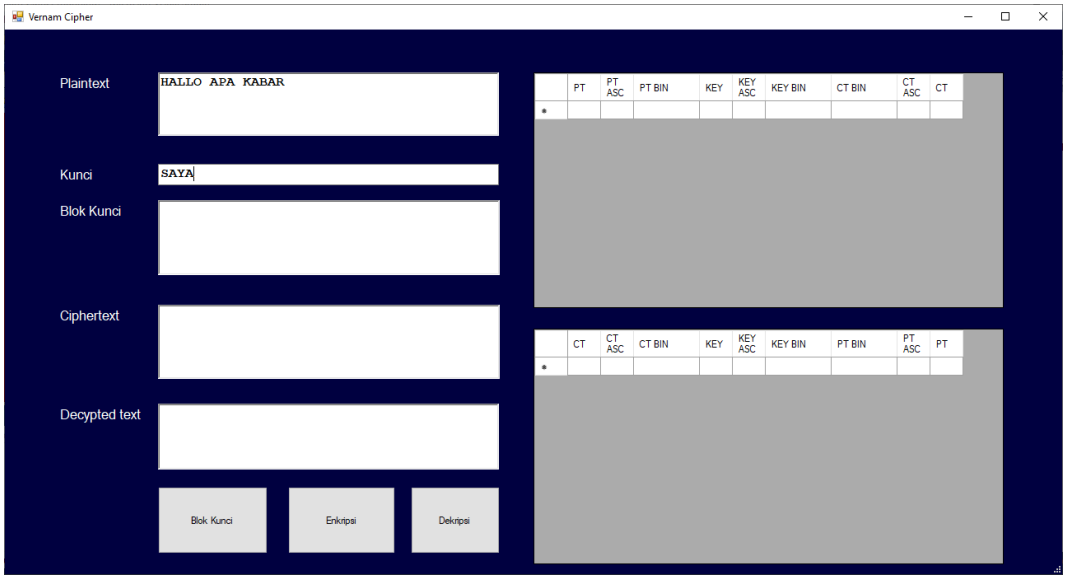
4.3.3 Memasukan Pesan



Gambar 4.3 Memasukan Pesan ke dalam plaintext

Gambar di atas menunjukan tampilan aplikasi, di dalam aplikasi ini terdapat Plaintext untuk memasukan pesan yang ingin dirahasiakan atau disandikan. Contoh pesan yang ingin disandikan di atas adalah “HALLO APA KABAR”.

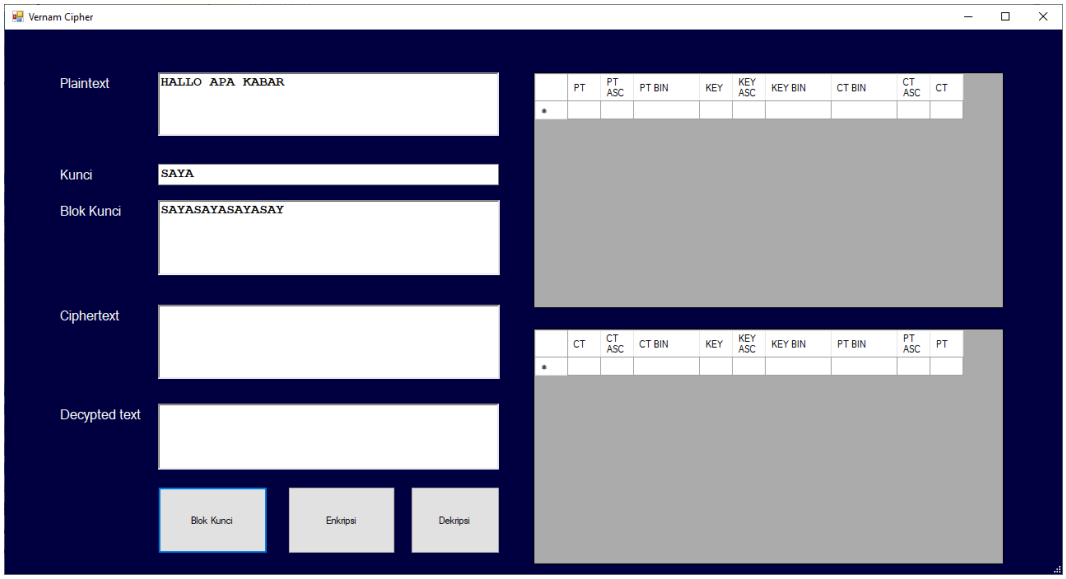
4.3.4 Memasukan Kunci



Gambar 4.4 Memasukan Kunci

Gambar di atas menunjukan tampilan aplikasi dimana terdapat bagian-bagian untuk memasukan kunci. Kunci tersebut adalah kunci yang ingin kita buat dan kunci tersebut sangan penting karena itulah sebagai pengaman sebuah pesan. Contoh kunci di atas adalah “SAYA”, kita juga dapat membuat kunci kita sendiri sesuai kemauan kita.

4.3.5 Klik Tombol Blok Kunci



Gambar 4.5 Blok Kunci

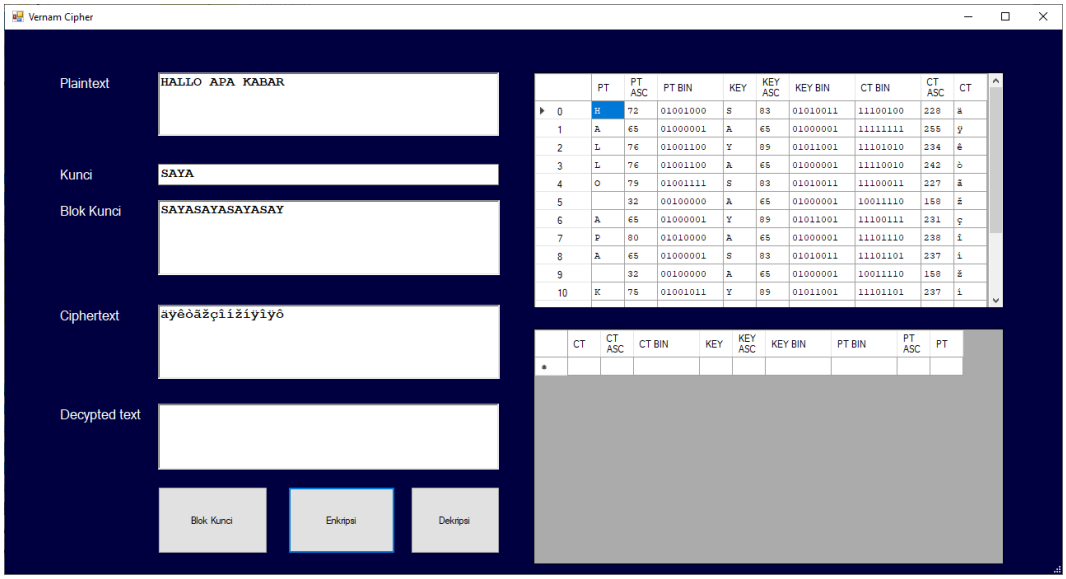
Gambar di atas menunjukan tampilan aplikasi yang terdapat Blok Kunci. Blok Kunci ini digunakan untuk memblok kuci yang ada di pilihan kunci, sehingga kunci yang jumlah hurufnya menyesuaikan dengan pesan yg ingin dirahasiakan. Blok Kunci akan mengulang kunci agar jumlah hurufnya sesuai dengan pesan yang akan disandikan. Contoh pada gambar di atas adalah:

Kuncinya = SAYA

Maka Blok Kunci akan memblok “SAYA” menjadi sesuai jumlah huruf pesan.

Blok Kunci = SAYASAYASAYASA

4.3.6 Klik Tombol Enkripsi



Gambar 4.6 Enkripsi

Gambar di atas menunjukan tampilan aplikasi yang terdapat tombol Enkripsi. Enkripsi berfungsi untuk menyandikan pesan yang sudah diisi di dalam kolom Plaintext menjadi pesan rahasia dan akan dimasukan kedalam kolom Ciphertext.

4.3.7 Hasil dari Enkripsi

	PT	PT ASC	PT BIN	KEY	KEY ASC	KEY BIN	CT BIN	CT ASC	CT
0	H	72	01001000	S	83	01010011	11100100	228	ä
1	A	65	01000001	A	65	01000001	11111111	255	ÿ
2	L	76	01001100	Y	89	01011001	11101010	234	ê
3	L	76	01001100	A	65	01000001	11110010	242	ò
4	O	79	01001111	S	83	01010011	11100011	227	ã
5		32	00100000	A	65	01000001	10011110	158	ž
6	A	65	01000001	Y	89	01011001	11100111	231	ç
7	P	80	01010000	A	65	01000001	11101110	238	î
8		32	00100000	S	83	01010011	10001100	140	Œ
9	K	75	01001011	A	65	01000001	11110101	245	õ
10	A	65	01000001	Y	89	01011001	11100111	231	ç
11	B	66	01000010	A	65	01000001	11111100	252	ü
12	A	65	01000001	S	83	01010011	11101101	237	í
13	R	82	01010010	A	65	01000001	11101100	236	ì
*									

Gambar 4.7 Hasil dari Enkripsi Pesan

Gambar diatas menunjukan tampilan hasil dari enkripsi pesan “HALLO APA KABAR” menjadi pesan yang sudah disandikan atau dirahasiakan. Pada gambar tersebut terdapat beberapa kolom:

PT = Plaintext

PT ASC = Plaintext ASCII

PT BIN= Plaintext Binner

KEY = Kunci

KEY ASC = Kunci ASCII

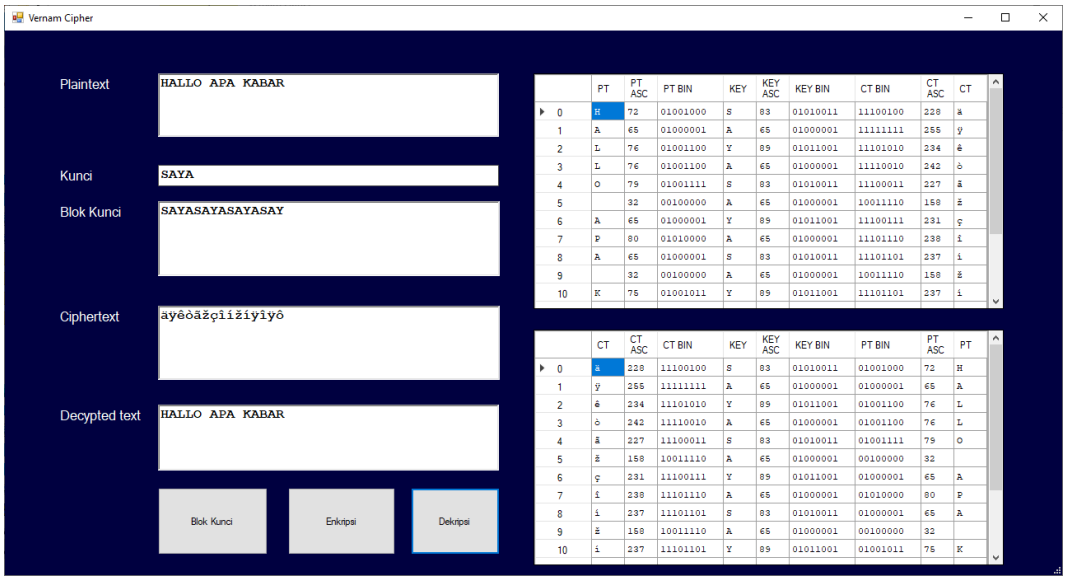
KEY BIN = Kunci Binner

CT BIN = Ciphertext Binner

CT ASC = Ciphertext ASCII

CT = Ciphertext

4.3.8 Klik Tombol Dekripsi



Gambar 4.8 Dekripsi

Gambar di atas menunjukan tampilan aplikasi yang terdapat tombol Dekripsi. Dekripsi berfungsi untuk mengembalikan pesan yang sudah disandikan menjadi pesan yang dapat dibaca kembali.

Kita juga dapat membuktikanya dengan cara manual:

Selanjutnya akan di enkripsi dengan formula Algoritma *Vernam Cipher* yaitu:

C = P + K

Dalam hal ini plaintext adalah *Ciphertext* 1 yang didapat.

CT1 = H xor S

= 72 xor 83

= 01001000 xor 01010011

= 00011011

= 27 (Escape)

Xnor = 01001000 xnor 01010011

= 11100100

= 228 (**ö**)

CT2 = A xor A

= 65 xor 65

= 01000001 xor 01000001

= 00000000

= 0 (Null)

Xnor = 01000001 xnor 01000001

= 11111111

= 255 (Nbsp)

CT3 = L xor Y

= 76 xor 89

= 01001100 xor 01011001

= 00010101

= 21 (NAK)

Xnor = 01001100 xnor 01011001

= 11101010

= 234 (**Û**)

CT4 = O xor A

= 79 xor 65

= 01001111 xor 01000001

= 00001110

= 14 (SO)

Xnor = 01001111 xnor 01000001

= 11110001

= 241 (**±**)

CT5 = SPACE xor S

= 32 xor 83

= 00100000 xor 01010011

= 01110011

= 115 (**s**)

Xnor = 00100000 xnor 01010011

= 10001100

= 140 (î)

CT6 = A xor A

= 65 xor 65

= 01000001 xor 01000001

= 00000000

= 0 (Null)

Xnor = 01000001 xnor 01000001

= 11111111

= 255 (Nbsp)

CT7 = P xor Y

= 80 xor 89

= 01010000 xor 01011001

= 00001001

= 9 (HT)

Xnor = 01010000 xnor 01011001

= 11110110

= 246 (÷)

CT8 = A xor A

= 65 xor 65

= 01000001 xor 01000001

= 00000000

= 0 (Null)

Xnor = 01000001 xnor 01000001

= 11111111

= 255 (Nbsp)

CT9 = SPACE xor S

= 32 xor 83

= 00100000 xor 01010011

= 01110011

= 115 (**s)**

Xnor = 00100000 xnor 01010011

= 10001100

= 140 (î)

CT10= K xor A

= 75 xor 65

= 01001011 xor 01000001

= 00001010

= 10 (LF)

Xnor = 01001011 xnor 01000001

= 11110101

= 245 (**§)**

CT11= A xor Y

= 65 xor 89

= 01000001 xor 01011001

= 00011000

= 24 (Cancel)

Xnor = 01000001 xnor 01011001

= 11100111

= 231 (**þ**)

CT12= B xor A

= 66 xor 65

= 01000010 xor 01000001

= 00000011

= 3 (ETX)

Xnor = 01000010 xnor 001000001

= 11111100

= 252 (³)

CT13= A xor S

= 65 xor 83

= 01000001 xor 01010011

= 00010010

= 18 (DC2)

Xnor = 01000001 xnor 01010011

= 11101101

= 237 (**Ý)**

CT14= R xor A

= 82 xor 65

= 01010010 xor 01000001

= 00010011

= 19 (DC3)

Xnor = 01010010 xnor 01000001

= 11101100

= 236 (**ý)**

Sehingga *Ciphertext* kedua yang didapat adalah:

Ciphertext = **ö** Nbsp **Û± s** Nbsp

4.3.9 Hasil dari Dekripsi

	CT	CT ASC	CT BIN	KEY	KEY ASC	KEY BIN	PT BIN	PT ASC	PT
0	ä	228	11100100	S	83	01010011	01001000	72	H
1	ÿ	255	11111111	A	65	01000001	01000001	65	A
2	ê	234	11101010	Y	89	01011001	01001100	76	L
3	ò	242	11110010	A	65	01000001	01001100	76	L
4	ã	227	11100011	S	83	01010011	01001111	79	O
5	ž	158	10011110	A	65	01000001	00100000	32	
6	ç	231	11100111	Y	89	01011001	01000001	65	A
7	î	238	11101110	A	65	01000001	01010000	80	P
8	í	237	11101101	S	83	01010011	01000001	65	A
9	õ	245	11110101	A	65	01000001	01001011	75	K
10	ç	231	11100111	Y	89	01011001	01000001	65	A
11	ü	252	11111100	A	65	01000001	01000010	66	B
12	í	237	11101101	S	83	01010011	01000001	65	A
13	ì	236	11101100	A	65	01000001	01010010	82	R
*									

Gambar 4.9 Hasil dari Dekripsi

Gambar diatas menunjukan hasil dari Ekripsi ke Dekripsi agar pesan tersebut dapat bisa dibaca kembali. Pada gamar tersebut terdapat beberapa kolom:

CT = Ciphertext

CT ASC = Ciphertext ASCII

CT BIN = Ciphertext Binner

KEY = Kunci

KEY ASC = Kunci ASCII

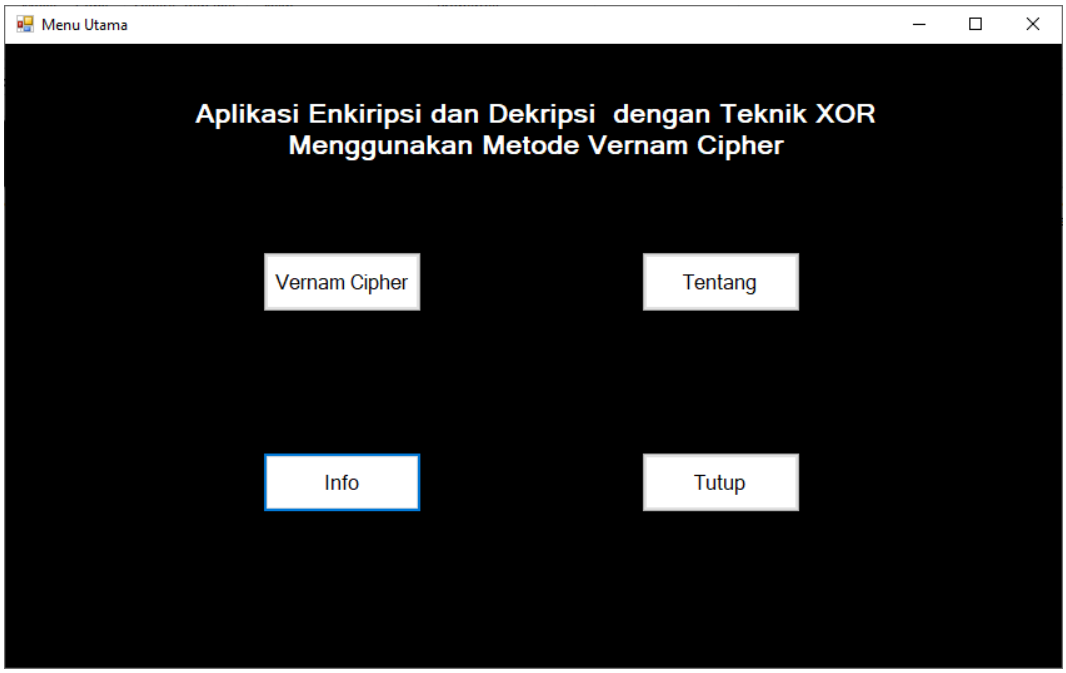
KEY BIN = Kunci Binner

PT BIN= Plaintext Binner

PT ASC = Plaintext ASCII

PT = Plaintext

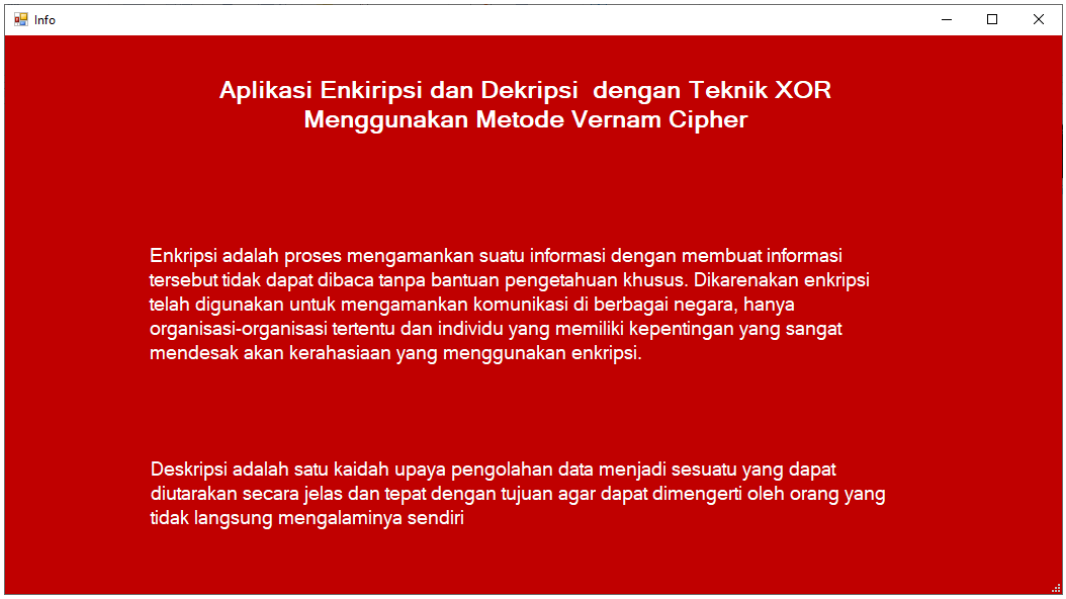
4.3.10 Menu Utama ke Info



Gambar 4.10 Menu Utama ke Info

Gambar di atas menunujukan tampilan menu utama kembali, ada beberapa menu lagi yaitu info, tentang dan tutup. Sekarang kita akan masuk ke menu Info.

4.3.11 Tampilan Info



Gambar 4.11 Info

Gambar di atas menampilkan info tentang Enkripsi dan Dekripsi. Pada gambar tersebut menjelaskan Enkripsi adalah proses mengamankan suatu informasi dengan membuat informasi tersebut tidak dapat dibaca tanpa bantuan pengetahuan khusus. Dekripsi dijelaskan sebagai suatu kaidah upaya pengelolahan data menjadi sesuatu yang dapat diutarakan secara jelas dan tepat dengan tujuan agar dapat dimengerti oleh orang yang tidak langsung mengalaminya sendiri.

4.3.12 Menu Utama ke Tentang

Gambar 4.12 Menu Utama ke Tentang

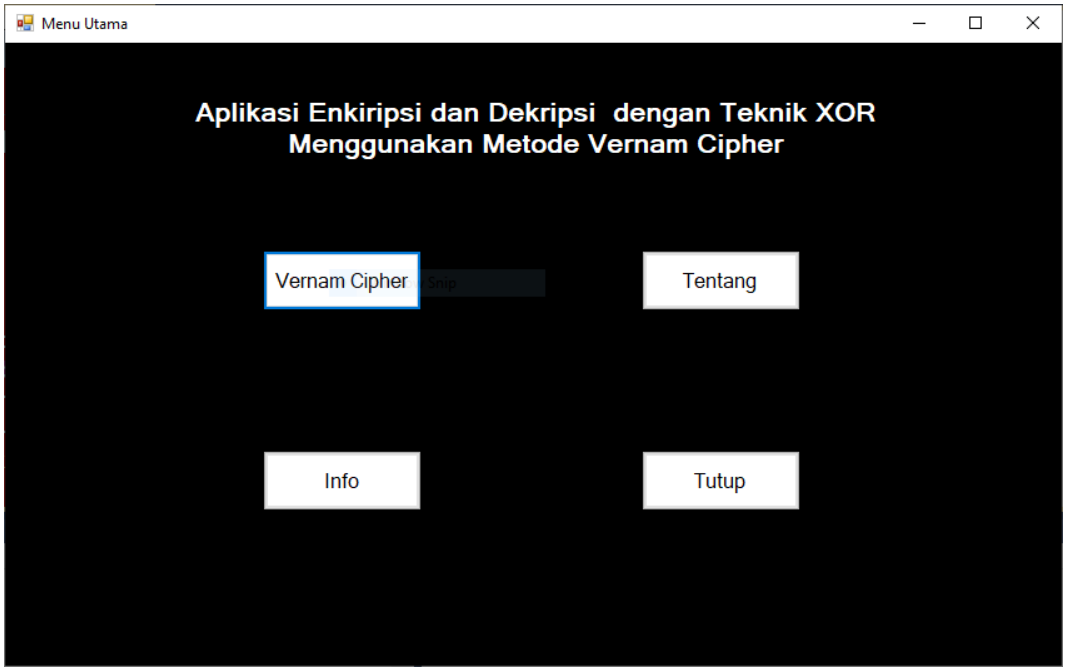
Gambar di atas adalah menu utama dan ada dua pilihan yang belu dibuka yaitu Tentang dan Tutup, Maka kita akan memilih Tentang

4.3.13 Tampilan Tentang Saya

Gambar 4.13 Tentang Saya

Gambar di atas menampilkan tentang profil saya sebagai mahasiswa Universita Pembangunan Pancabudi Medan.

4.3.14 Menu Utama Ke Tutup



Gambar 4.14 Menu Utama ke Tutup

Gambar di atas merupakan Menu Utama yang kita belum klik yaitu menu Tutup, agar menyudahi aplikasi ini kita klik Tutup.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Setelah melakukan perancangan Aplikasi Enkripsi dan Dekripsi dengan Teknik XOR menggunakan Metode Vernam Cipher, dapat diperoleh kesimpulan berdasarkan pengamatan yang dilakukan penulis selama proses pengerjaan skripsi ini antara lain sebagai berikut :

1. Integritan pesan akan terjaga yaitu dengan keamanan pesan yang membuat pesan tidak dapat dibaca atau diartikan oleh pihak lain.
2. Dengan adanya perancangan Aplikasi Enkripsi dan Dekripsi dengan Teknik XOR menggunakan Metode Vernam Cipher ini dapat mempermudah untuk mengirimkan pesan rahasia

5.2 Saran

Saran yang dapat diberikan penulis untuk menyempurnakan skripsi ini adalah sebagai berikut:

1. Sistem ini diharapkan dapat menjawab kebutuhan akan keamanan pesan yang dikirim maupun diterima seseorang.
2. Sistem aplikasi ini diharapkan dapat dikembangkan dengan metode kriptografi lain, yang mempunyai spesifikasi keamanan yang lebih tinggi tanpa merusak intergritas pesan atau data.

3. Dalam pengembangan kedepan sangatlah penting untuk memperhatikan dan meningkatkan tingkat keamanan yang lebih baik lagi untuk menjaga keamanan pesan yang penting, sehingga tidak sembarangan orang dapat mengakses pesan tersebut.
4. Untuk lebih meningkatkan kenyamanan pemakaian aplikasi, perlu dilakukan pengembangan desain dan fitur-fitur yang lebih banyak.

DAFTRA PUSTAKA

Badawi, A. (2018). Evaluasi Pengaruh Modifikasi Three Pass Protocol Terhadap Transmisi Kunci Enkripsi.

Bahri, S. (2019). Optimasi Cluster K-Means dengan Modifikasi Metode Elbow untuk Menganalisis Disrupsi Pendidikan Tinggi.

Dony Ariyus. (2006). Pengantar Ilmu Kriptografi Teori, Analisis dan Implementasi. Buku.

Diantoro, M., Maftuha, D., Suprayogi, T., Iqbal, M. R., Mufti, N., Taufiq, A., ... & Hidayat, R. (2019). Performance of Pterocarpus Indicus Willd Leaf Extract as Natural Dye TiO2-Dye/ITO DSSC. Materials Today: Proceedings, 17, 1268-1276.

Dhany, H. W., Izhari, F., Fahmi, H., Tulus, M., & Sutarman, M. (2017, October). Encryption and decryption using password based encryption, MD5, and DES. In International Conference on Public Policy, Social Computing and Development 2017 (ICOPOSDev 2017) (pp. 278-283). Atlantis Press.

Eko Hari Rachmawanto, Christy Atika Sari, Yani Parti Astuti, Liya Umaroh. (2016). Kriptografi Vernam Cipher Untuk Mencegah Pencurian Data Pada Semua Ekstensi File. Diakses dari https://www.unisbank.ac.id/ojs/index.php/sendi_u/article/view/4164

Fresly Nandar Pabokory, Indah Fitri Astuti, Awang Harsa Kridalaksana. (2015). Implementasi Kriptografi Pengamanan Data Pada Pesan Teks, Isi File Dokumen, Dan File Dokumen Menggunakan Algoritma Advanced Encryption Standard. *Journal* Informatika Mulawarman. Diakses dari https://www.researchgate.net/publication/323962079_Implementasi_Kriptografi_Pengamanan_Data_Pada_Pesan_Teks_Isi_File_Dokumen_Dan_File_Dokumen_Menggunakan_Algoritma_Advanced_Encryption_Standard

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

Gun Gun Maulana. (2017). Pembelajaran Dasar Algoritma Dan Pemrograman Menggunakan El-Goritma Berbasis Web. *Journal* Teknik Mesin (JTM). Diakses dari https://media.neliti.com/media/publications/177019-ID-pembelajaran-dasar-algoritma-dan-pemrogr.pdf

Hariyanto, E., Lubis, S. A., & Sitorus, Z. (2017). Perancangan prototipe helm pengukur kualitas udara. KOMIK (Konferensi Nasional Teknologi Informasi dan Komputer), 1(1).

Hariyanto, E., Iqbal, M., Siahaan, A. P. U., Saragih, K. S., & Batubara, S. (2019, March). Comparative Study of Tiger Identification Using Template Matching

Approach based on Edge Patterns. In Journal of Physics: Conference Series (Vol. 1196, No. 1, p. 012025). IOP Publishing.

Iqbal, M., Siahaan, A. P. U., Purba, N. E., & Purwanto, D. (2017). Prim's Algorithm for Optimizing Fiber Optic Trajectory Planning. Int. J. Sci. Res. Sci. Technol, 3(6), 504-509.

Mohammad Jumeidi, Dedi Triyanto, Yulrio Brianorman. (2016). Implementasi Algoritma Kriptografi Vernam Cipher Berbasis Fpga. *Journal* Coding. Diakses dari https://id.scribd.com/document/357391240/Impelementasi-Algoritma-Vernam-Chiper-Berbasis-FPGA

Rifki Sadikin. (2012). Kriptografi Untuk Keamanan Jaringan. Buku.

Rahim, R., & Fuad, R. N. (2019). Aplikasi dalam simulasi penjualan dengan menggunakan metode monte carlo. Ready Star, 2(1), 235-239.

Rahim, R., & Fuad, R. N. (2019). Aplikasi dalam simulasi penjualan dengan menggunakan metode monte carlo. Ready Star, 2(1), 235-239.

Ramadhan, Z., Zarlis, M., Efendi, S., & Siahaan, A. P. U. (2018). Perbandingan Algoritma Prim dengan Algoritma Floyd-Warshall dalam Menentukan Rute Terpendek (Shortest Path Problem). JURIKOM (Jurnal Riset Komputer), 5(2), 135-139.

Sitepu, N. B., Zarlis, M., Efendi, S., & Dhany, H. W. (2019, August). Analysis of Decision Tree and Smooth Support Vector Machine Methods on Data Mining. In Journal of Physics: Conference Series (Vol. 1255, No. 1, p. 012067). IOP Publishing.

Sumartono, I. (2019). Analisis Perancangan Sistem Rencana Pembelajaran Terpadu dalam Mendukung Efektivitas dan Mutu Pengajaran Dosen (Studi Kasus: Fakultas Ilmu Komputer Universitas Pembangunan Panca Budi). Jurnal Teknik dan Informatika, 6(1), 12-17.

Sitorus, Z., & Siahaan, A. P. U. (2016). Heuristic Programming in Scheduling Problem Using A* Algorithm. IOSR J. Comput. Eng, 18(5), 71-79.

Sharif, A. (2019). data mining untuk memprediksi itemset promosi penjualan barang menggunakan metode market basket analysis (mba)(studi kasus: toko sentra ponsel). Jurnal Mantik Penusa, 3(2, Des).

Utomo, R. B. (2019). Aplikasi Pembelajaran Manasik Haji dan Umroh berbasis Multimedia dengan Metode User Centered Design (UCD). J-SAKTI (Jurnal Sains Komputer dan Informatika), 3(1), 68-79.

Wahyuni, S., Lubis, A., Batubara, S., & Siregar, I. K. (2018, September). implementasi algoritma crc 32 dalam mengidentifikasi keaslian file. In Seminar Nasional Royal (SENAR) (Vol. 1, No. 1, pp. 1-6).