

## Implementasi Enkripsi Dekripsi Pesan Teks Menggunakan Model Julis Caesar Berbasis Object Oriented Programme

Reditya Kiko Pandu Pratama<sup>1</sup>, Fitri Latifah<sup>2</sup>  
Sekolah Tinggi Manajemen Informatika dan Komputer Nusa Mandiri  
Program Studi Teknik Informatika  
Jl. Kramat Raya 25  
[redityakiko@yahoo.co.id](mailto:redityakiko@yahoo.co.id)  
[Fitri.flr@bsi.ac.id](mailto:Fitri.flr@bsi.ac.id)

### ABSTRACT

*current data and information becomes a thing that cannot be separated again. But at a certain moment we forget the essential aspect, namely the security aspects of such information and data. Therefore we must always be vigilant in terms of data security. Encryption techniques and the description of which writer will apply message text, the security techniques adopted from ancient security techniques, which writer apply to the programming language. Change from plain text to cipher text is a basic function of encryption techniques and this description. Encryption techniques and my description, using the algorithm of Julius Caesar was able to secure data.*

**Keywords:** *Julius Caesar, Encryption, Decryption, Message Text*

### PENDAHULUAN

Perkembangan teknologi saat ini memungkinkan manusia untuk berkomunikasi dan saling bertukar informasi dengan jarak jauh. Pertukaran informasi jarak jauh seperti antar kota, antar wilayah dan bahkan antar benua sudah tidak menjadi kendala lagi namun dilain hal keamana atau sekuritas terhadap kerahasiaan informasi saat inilah yang menjadi persolan.

Banyak pengguna internet yang saat ini khawatir terhadap pengirima informasi diketahui oleh pihak lain. Untuk mengatasi persoalan ini sebuah metode untuk keamanan informasi yang dikenal dengan kriptografi. Dalam kriptografi dikenal dengan dua konsep utama yakni enkripsii dan dekripsi .

Pada penelitian ini penulis mempunyai tujuan untuk mengkaji dan

membangun sebuah aplikasi enkripsi dan dekrpsi pesan text dengan menggunakan algorithma *Julius Casear* berbasiskan *Objeck Oriented Progam*

### BAHAN DAN METODE

Menurut Latif (2010 : 1) dalam penelitiannya terdapat dua proses penting di dalam kriptografi yang berperan dalam merahasilakan suatu informasi yaitu enkripsi (encryption) dan dekripsi (decription). Enkripsi adalah bentuk transformasi datas (plaintext) ke dalam bentuk yang hamper tidak dapat dibaca (ciphertext) tanpa pengetahuan yang cukup.

Tujuan dari enkripsi adalah untuk menjamin kerahasiaan dengan menjaga informasi tersembunyi dari siapapun yang bukan pemilik atau memiliki akses terhadap data yang telah dienkrpsi.

Sedangkan dekripsi adalah kebalikan dari enkripsi yaitu transformasi dari data yang telah dienkripsi (ciphertext) kembali ke bentuk semula (plaintext). Proses enkripsi dan dekripsi pada umumnya membutuhkan penggunaan sejumlah informasi rahasia, yang sering disebut dengan kunci (key)

Menurut Sudrajat (2006 :18) bahwa transparent data encryption menawarkan kemudahan dari sisi pengguna sehingga pengguna dapat langsung melakukan proses enkripsi data tanpa melakukan coding dan kompleksitas manajemen key.

## A. Konsep Dasar Program

### a. Program

Program, menurut Binanto (2005 :1) adalah:

1. Untuk mendeskripsikan instruksi – instruksi tersendiri yang biasanya disebut dengan source code yang dibuat oleh programmer
2. Untuk mendeskripsikan suatu keseluruhan bagian dari software yang executable

Bedasarkan dua pernyataan diatas bahwa bahasa yang dapat dimengerti oleh mesin computer disebut dengan bahasa pemrograman.

Menurut Binanto (2005:1) pemrograman dapat diartikan sebagai cara membuat program dalam konteks ini berarti membuat program computer atau pemrograman adalah suatu kumpulan urutan perintah ke computer untuk mengerjakan sesuatu.

Sedangkan menurut Yulikuspartono (2004:2) kegiatan menulis atau membuat langkah – langkah instruksi tersebut dalam suatu bahasa komputer dikenal dengan nama pemrograman dan hasil yang telah dicapai oleh penulisan tersebut dikenal dengan program.

## 1.1. Algoritma

### a. Algoritma

Menurut Yulikuspartono (2004:2) usaha yang dilakukan agar solusi suatu masalah dapat diperoleh adalah membuat suatu proses atau prosedur yang merupakan urutan dan langkah – langkah atau instruksi – instruksi yang terintegrasi, langkah – langkah instruksi tersebut secara umum kita sebut dengan Algoritma. Dalam kamus Webster Dictionary dalam Yulikuspartono (2004:2) Algoritma adalah suatu metode khusus yang digunakan untuk menyelesaikan suatu masalah yang nyata. Oleh karena itu algoritma pada hakekatnya merupakan suatu prosedur yang tepat untuk dapat memecahkan masalah dengan menggunakan bantuan computer serta bahasa pemrograman tertentu.

### b. Kriteria Pemilihan Algoritma

Kriteria atau sifat – sifat algoritma yang terbaik menurut Yulikuspartono (2004:3) adalah:

1. Ada Output  
Mengacu pada definisi algoritma maka suatu algoritma haruslah mempunyai solusi dari masalah yang sedang diselesaikan
2. Efektivitas dan Efisiensi  
Dikatakan efektif jika algoritma tersebut menghasilkan suatu solusi yang sesuai dengan masalah yang diselesaikan dalam arti algoritma harus tepat guna, sedangkan yang dimaksud dengan efisiensi adalah jika waktu proses suatu algoritma relative lebih singkat dan penggunaan memori komputernya lebih sedikit.
3. Jumlah Langkahnya berhingga  
Hal ini dapat diartikan bahwa barisan instruksi yang dibuat harus dalam suatu urutan tertentu atau harus terhingga agar masalah yang dihadapi dapat diselesaikan dengan tidak memerlukan waktu yang lama
4. Berakhir  
Proses didalam mencari penyelesaian suatu masalah harus berhenti dan berakhir dengan hasil akhir yang

merupakan solusi berupa informasi yang tidak diketemukan solusinya

#### 5. Terstruktur

Adalah urutan barisan langkah – langkah yang digunakan harus disusun sedemikian rupa proses penyelesaian tidak berbelit sehingga bagian proses dapat dibedakan dengan jelas mana bagian input, proses dan output sehingga memudahkan pengguna melakukan pemeriksaan ulang.

### 1.2. Kriptografi

Menurut Ariyus (2006:9) kata kriptografi berasal dari bahasa Yunani, sedangkan menu bahasa kriptografi dibagi menjadi dua yaitu kriptografi dan graphia, kriptografi berarti rahasia sedangkan graphia adalah tulisan. Jadi kriptografi adalah ilmu dan seni untuk menjaga keamanan pesan ketika pesan dikirim dari suatu tempat ke tempat yang lain.

### 1.3. Algoritma Kriptografi

Menurut Ariyus (2006:13) Algoritma kriptografi terdiri dari tiga fungsi dasar yaitu

1. Enkripsi. Enkripsi merupakan hal yang sangat penting dalam kriptografi yang merupakan pengamanan data yang dikirimkan terjaga rahasianya. Pesan asli disebut plaintext yang dirubah menjadi kode – kode yang tidak dimengerti. Enkripsi dapat diartikan dengan cliper atau kode.
2. Deskripsi, merupakan merupakan kebalikan dari enkripsi
3. Kunci adalah alat yang digunakan untuk melakukan enkripsi dan dekripsi, kunci terbagi menjadi duabagian yaitu kunci pribadi (private key) dan kunci umum (public key).

### 1.4. Macam – macam Algoritma Kriptografi

#### a. Algoritma Simetri

Menurut Ariyus (2006 :14) Algoritma Simetris adalah algoritma klasik karena memakai kunci yang sama untuk kegiatan enkripsi dan dekripsinya, menurut Ariyus juga (2006:4) algoritma yang memiliki kunci simetri diantaranya adalah :

1. Data Encryption Standart (DES),
2. RC2, RC4, RC5, RC6,
3. International Data Encryption Standar (IDEA),
- 4 Advance Encryption Standar (AES),
- 5 One Time Pad (OTP), (6) A5

#### b. Algoritma Asimetri

Menurut Ariyus (2006:15) Algoritma asimetri sering juga disebut dengan algoritma kunci public, dengan kata lain kunci yang digunakan untuk melakukan enkripsi dan dekripsi berbeda., pada algoritma asimetri kunci terbagi menjadi dua bagian yakni :

(a) Kunci umum (public key) kunci yang boleh semua orang tahu (dipublikasikan ) algoritma yang menggunakan kunci ini adalah :

1. Digital Signature Algoritma (DSA),
2. RSA
- 3 Diffie Hellman (DH),
- 4 Elliptic Curve Cryptography (ECC).

(b) Kunci pribadi ( Private key) adalah kunci yang dirahasiakan ( hanya boleh diketahui oleh satu orang).

### 1.5. Kriptografi Klasik

Menurut Ariyus (2006:16) kriptografi klasik merupakan suatu algoritma yang menggunakan satu kunci untuk menggunakan data, teknik ini sudah digunakan beberapa abad yang lalu.

Dua teknik dasar yang dapat digunakan untuk algoritma jenis ini diantaranya adalah : (1) Teknik Substitusi pengantian setiap karakter plaintext karakter lain (2)

Teknik Transportasi (permutasi) teknik ini menggunakan permutasi karakter.

**1.6. Caesar Chiper**

Menurut Ariyus (2006:16) substitusi chipper yang pertama dalam dunia penuadidan padawaktu pemerintahan Yulius Caesar dikenal dengan Caesar Chiper dengan menggantikan pposisi huruf awal dari alphabet. Algorithma Caesar dapat dipecahkan dengan metode Caesar chipper, menurut Ariyus (2006:18) Caesar cripher dapat terpecahkan dengan metode kemungkinan – kemungkinan untuk menemukan kunci sampai kunci itu ditemukan. Banyaknya kunci dari sautu cpher, cukup meresponkan para cryptanalysis, walaupun dicoba dengan cara Brute Forces Attack akan membutuhkan waktu untuk menemukan kunci dari cipertext. Tapi cara ini selalu berhasil walaupun memerlukan waktu yang lama.

**Tabel 1**  
**Sebelum di Enkripsi Dengan Caesar Cipher**

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25

Sumber : (Ariyus:17)

**Tabel II.2.**  
**Sesudah di Enkripsi Dengan Caesar Cipher**

D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	1	2	3

Sumber : (Ariyus:17)

**1.7. Crytanalysis**

Menurut Ariyus (2006:198) secara umum diasumsikan bahwa cryptanalysis adalah kebalikan dari kriptosistem yang dipakai. Menurut Ariyus (2006:199) secara umum crytanalysisi mencoba untuk menembus kerahasiaan dari suatu kode. Mereka mencoba membangun suatu algoritma untuk memecahkan kode dari ciphertext, secara umum urutan dari algorithma nya sebagai berikut :

1. Information Constrain  
Para cryptanalysis harus mendapatkan salah satu dari komponen kode itu sendiri untuk memulai suatu serangan
2. Ciphertext Only  
Crytanalysisi hanya memiliki pesan ciphertext semuanya dienkrip dengan algorithma yang sama Crytanalysis tidak mengetui kunci dan plantextnya.
3. Know  
Plaintext dimana kemungkinan crytanalysisi mengetahui beberapa plaintext dan juga ciphertextnya
4. Defined Plaintext  
Crytanalysis tidak hanya mengetahui beberapa plaintext dan ciphertext namun bebas memilih plaintext yang dienkripsi dengan algorithma yang sama.

Adapun langkah yang digunakan untuk melakukan pengujian system dengan metode Julius Caesar adalah sebagai berikut:

1. Isi pesan teks dikotak plaintext
2. Jika pesan teks tidak di isi di kotak plaintext maka ada pesan pemberitahuan bahwa pesan harus di isi
3. Klik tomol ekripsi untuk menengkrip pesan teks tersebut
4. Kemudian hasil enkripsi akan tampil akan tampil di kotak chiptext

5. Jika pesan teks tidak di isi di kotak chiptext maka ada pean pemberitahuan bahwa pesan harus di isi
6. Klik tombol deskripsi untuk mengembalikan pesan teks yang terenkripsi menjadi pesan teks seperti semula
7. Tombol clear untuk menghapus dan membuat pesan teks yang baru akan denkripsi atau di dekripsikan
8. Tombol close untuk keluar dari aplikasi tersebut
9. Menu bar file berisi menu enkripsi, Deskripsi dan save
10. Menu baru About me merupakan informasi singkat dari perancangan aplikasi tersebut.

### 1.8. Peralatan Pendukung

Pada penelitian akan dijelaskan tools yang digunakan untk menjaelaskan aplikasi yang akan dibangun

#### a. Object Orented Programming (OOP)

Menurut Sholiq (2006:21) Object Orieted merupakan paradigma baru dalam rekayasa perangkat lunak yang menmandang sistem sebagai kumpulan objek objek diskrit yang saling berinteraksi. Yang dimaksud berorientasi objek adalah bahwa mengorganisasikan perangkat lunak sebagai kumpulan objek objek yang bekerja sama antara informasi atau struktur data dan perilaku (behavior) yang mengaturnya.

Menurut Sholiq (2006:22) bahwa: Pada pemrograman, berorientasi obyek didefinisikan sebagai suatu cara membuat program yang mempunyai beberapa keuntungan. Ia menggunakan pendekatan component-based untuk pengembangan perangkat lunak, dimana pertama kali dimulai dengan membuat sebuah sistem yang merupakan kumpulan obyek-obyek. Kemudian saat pengembangan sistem, dilakukan dengan menambahkan komponen-komponen ke

sistem atau dengan cara menciptakan komponen baru dari komponen yang sudah ada. Pada akhirnya, jika membangun sistem baru, maka cukup dengan menggunakan kembali obyek-obyek yang telah dibuat ke sistem baru yang sedang dibangun.

Menurut Bahri, Sjachriyanto (2005:85) Keuntungan bahasa pemrograman OOP dibanding bahasa pendahulunya adalah:

1. Lebih terstruktur.
2. Lebih modular, serta mudah digunakan kembali (reusable).
3. Lebih tahan terhadap perubahan.
4. Lebih mudah dikembangkan.
5. Lebih “alami”.

Menurut Bahri, dkk (2005:85) Ciri dasar Object Oriented Programming adalah:

1. Classes (Kelas Objek), adalah definisi data dan fungsi/procedure yang terbungkus secara abstrak. Konsep pendefinisian kelas ini biasanya disebut sebagai Data Abstraction (Abstraksi Data). Definisi kelas ini biasanya akan menjadi kerangka modularitas program.
2. Encapsulation (Pembungkusan), adalah membungkus/menyembunyikan bagian-bagian tertentu dari program, sehingga pemakai objek tidak perlu tahu bagian internal dari objek tersebut.
3. Inheritance (Pewarisan), yaitu hubungan antar object yang memungkinkan disusunnya hirarki object. Setiap objek anak akan memiliki sifat dari moyangnya.
4. Polymorphism (keberakrupan kemampuan mengungkapkan atau melakukan hal yang sama dengan cara yang berbeda

#### 1.9. Unified Modeling Language (UML)

Menurut Nugroho (2010:6) "UML (Unified Modeling Language ) adalah bahasa permodelan untuk sistem atau perangkat lunak yang berparadigma berorientasi objek".

#### 1. Diagram Use Case

Menurut Sholih (2006:7) bahwa:Diagram use case atau use case diagram menyajikan interaksi antara use case dan aktor. Dimana, aktor dapat berupa orang, peralatan, atau sistem lain yang berinteraksi dengan sistem yang sedang dibangun. Use case menggambarkan fungsionalitas sistem atau persyaratan-persyaratan yang harus dipenuhi system dari pandangan pemakai.

#### 2. Diagram Aktivitas

Menurut Sholih (2006:8) bahwa:Diagram aktivitas atau activity diagram menggambarkan aliran fungsionalitas sistem. Pada tahap pemodelan bisnis, diagram aktivitas dapat digunakan untuk menunjukkan aliran kerja bisnis (business work-flow). Dapat juga digunakan untuk menggambarkan aliran kejadian (flow of events) dalam use case.

#### 3. Diagram Sekuensial

Menurut Sholih (2006:9) "Diagram sekuensial atau sequence diagram digunakan untuk menunjukkan aliran fungsionalitas dalam use case".

#### 4. Diagram Deployment

Menurut Sholih (2006:17) "Diagram deployment atau deployment diagram menampilkan rancangan fisik jaringan dimana berbagai komponen akan terdapat di sana. Pada sistem ATM, terdapat banyak subsistem yang dijalankan pada peralatan fisik yang terpisah atau sering disebut node.

#### 1.10. Komponen Desain Berorientasi Objek

Menurut Al Bahri (2006: 180) Desain berorientasi objek terdiri dari 4 (empat) komponen, yaitu :

##### a. Komponen Domain Permasalahan

Komponen ini berfungsi mengelola penggunaan class dan objek berupa system yang didesain seperti pada sebuah system pendaftaran siswa di sini objek yang diolah adalah siswa, guru dan mata pelajaran serta nilai

##### b. Kompoen Interaksi dengan Pengguna

Pada komponen ini berisi objek yang akan diproses, didisplay yang diinput dan digunakan oleh system untuk menghubungkan pengguna dengan system, seperti contohnya pada system berbasis windows untuk desain proses input data menggunakan teksbox, command dan frame, radio button.

##### c. Komponen manajemen Penugasan

Komponen ini berfungsi untuk mengkoordinasikan berbagai program yang terdapat di dalam suatu aplikasi

##### d. Komponen manajemen data

Komponen ini berfungsi bagaimana cara pengaksesan data serta melakukan manipulasi data yang tersimpan didalam database manajemen system .

## HASIL DAN PEMBAHASAN

### 1.11. Desain Algorithma

Algorithma Julius Caesar adalah varian dari bentuk algorithma populer pada zamannya, algorithma ini adalah salah satu algorithma yang digunakan untuk menyelesaikan masalah dengan menggunakan data berupa pesan teks yang akan di enkripsi, kemudian pesan yang di enkripsi tersebut dapat dikembalikan seperti semula dengan cara di deskripsi. Untuk menganalisis kebutuhan di atas kita menggunakan

kunci dengan nilai lima seperti dibawah ini

$$C=E(P)=(P+5) \text{ mod } (26)$$

Dan rumus deskripsinya adalah sebagai berikut :

$$P=D(C)=(C-K) \text{ mod } (26)$$

Dengan memasukan kunci lima angka maka formula menjadi :

$$P=D(C)=(C-5) \text{ mod } (26)$$

Jika kita menggunakan plaintext seperti dibawah ini

NUSA MANDIRI ADALAH KAMPUSKU

Dengan menggunakan kunci lima maka akan terbentuk ciphertext sebagai berikut :

SZXFRFSINWNFIPQFMPFRUZXPZ

### 1.12. Software Architecture

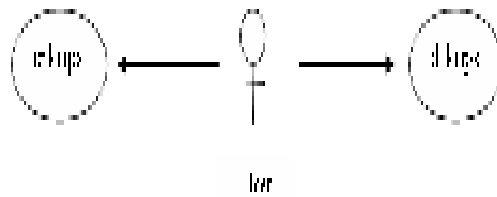
a. Pseudocode Algoritma Julius Caesar

```

procedure TForm1 enkripsiClick(Sender: TObject);
//enkripsi
teks[j] := chr(ord(teks[j])+5);

procedure TForm1 deskripsiClick(Sender: TObject);
//deskripsi
teks[j] := chr(ord(teks[j]-5));
    
```

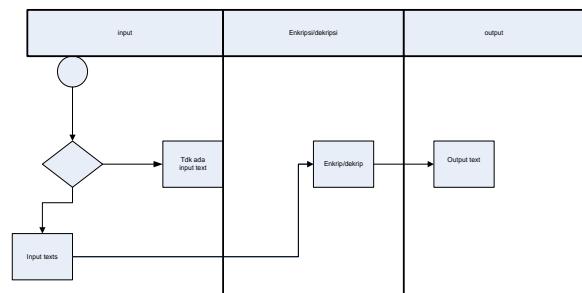
b. Usecase Diagram



Sumber : Pribadi

Gambar : 3.1. Use Case enkripsi

c. Activity Diagram

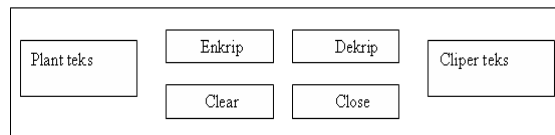


Sumber : Pribadi

Gambar : 3.2. Activity Diagram

d. User Interface

Rancangan dari tampilan atau *interface* yang berguna untuk membantu interaksi terhadap pengguna. Tampilan tersebut haruslah dijelaskan fungsi-fungsinya agar *user* dapat dengan mudah menggunakan aplikasi yang dimiliki pada rancangan *interface* tersebut. Gambar dibawah ini adalah *form* tampilan utamanya

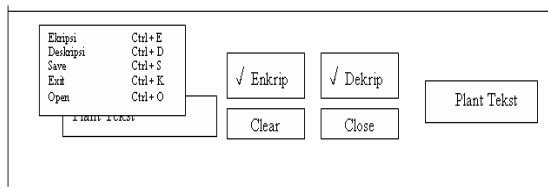


Sumber : Pribadi

Gambar .33. User Interface

Dibawah ini merupakan keterangan dari rancangan interface form tampilan utamanya yaitu :

1. Frame bagian atas adalah judul atau nama aplikasi yang dibuat.
2. Dibawah frame merupakan menu bar
3. Menu bar tersebut ada menu file dan about me
4. Kotak memo plaintext : Untuk menulis atau menampilkan pesan teks asli.
5. Kotak memo chipertext : Untuk menulis atau menampilkan pesan teks yang telah dienkrip.
6. Tombol enkripsi: Untuk mengenkripsi pesan teks dari plaintext ke chipertext.
7. Tombol dekripsi: Untuk mendekripsikan pesan teks dari chipertext ke plaintext.
8. Tombol clear: Untuk membuat pesan teks yang baru.
9. Tombol clear untuk membuat pesan baru
10. Tombol close: Untuk menutup aplikasi.



Sumber Pribadi

Gambar 3.2 User Interface

Dibawah ini merupakan keterangan dari rancangan interface menu bar file yaitu :

1. Di dalam menu file terdapat menu enkripsi dengan tombol shortcut CTRL + E, menu dekripsi dengan tombol shortcut CTRL + D, menu save dengan tombol shortcut CTRL + S.
2. Menu enkripsi dengan shortcut CTRL + E : Untuk mengenkripsi pesan teks dari plaintext ke chipertext.
3. Menu dekripsi dengan shortcut CTRL + D : Untuk mendekripsikan pesan teks dari chipertext ke plaintext.

4. Menu save dengan shortcut CTRL + S: Untuk menyimpan pesan teks dari plaintext maupun dari chipertext.

### Spesifikasi Hardware dan software

Tabel 3.1

Spesifikasi Hardware dan software

Kebutuhan	Keterangan
Sistem Operasi	: Windows 7
Processor	: Intel Pentium processor T4400 2.2 GHz, 800 MHz FSB
RAM	: 1 GB Memory
Harddisk	: 320 GB HDD
DVD ROM	: Super Multi DL Drive
Monitor	: 14 Ina HD LED LCD
Keyboard	: 108 Key
Printer	: Laser Jet
Software	: Borland Delphi 7, Microsoft Word 2007

### Algoritma Enkripsi

prosedure Form1.enkripsi

variabel teks = string

i = integer

mulai

jika Plaintext.Text <> " maka

mulai

teks := Plaintext.text

untuk i := 1 to length(teks) do

mulai

teks[i] + 5

jika teks[i] = '{' then

teks[i] := 'a' ;

jika teks[i] = '[' maka

teks[i] := 'A' ;

jika teks[i] = '|' maka

teks[i] := 'b' ;

jika teks[i] = '\' maka

teks[i] := 'B' ;

jika teks[i] = '}' maka

teks[i] := 'c' ;

jika teks[i] = 'w' maka

teks[i] := 'C' ;

berakhir;

chipertext.text := teks;

plaintext.Text:= ";

chipertext.SetFocus;

berakhir

jika yang lain mulai

MessageDlg('Plaintext harus diisi

!!!',mtInformation,[mbOK],0);

Plaintext.SetFocus;

berakhir

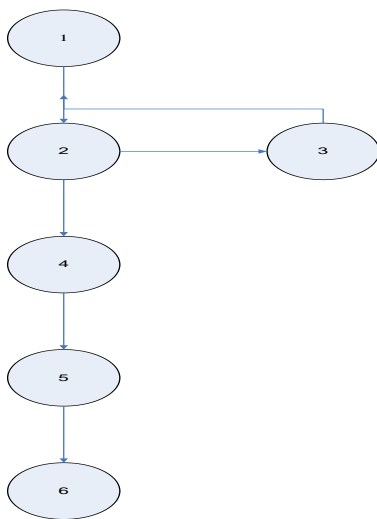
berakhir



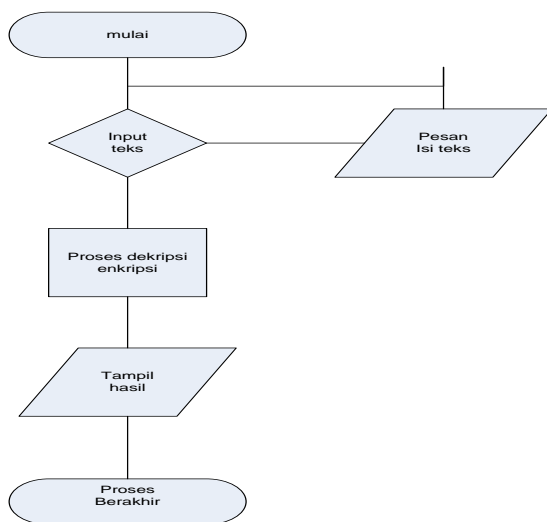
**Testing**

Pengujian disini bertujuan untuk mengetahui fungsionalitas algoritma pemrograman yang dirancang, apakah berjalan dengan baik atau tidak. Pada Pengujian ini peneliti menggunakan metode black box testing, white box testing dan acceptance testing. Black box testing bertujuan untuk menunjukkan fungsi perangkat lunak tentang cara beroperasinya, apakah pemasukan data keluaran telah berjalan sebagaimana yang diharapkan. White box testing adalah meramalkan cara kerja perangkat lunak secara rinci, oleh karena itu jalur logika akan di tes setidaknya satu kali.

a. Pengujian White box untuk jalur logika 1(true) dan 0 (false)



Gambar 4.1 Pengujian White Box



Gambar 4.2 Flowchart Pengujian

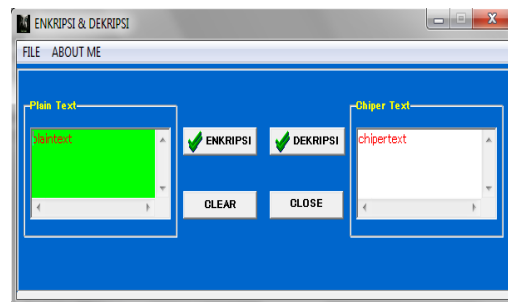
b. Pengujian Black Box

**Tabel: 4.1. Hasil Pengujian Black Box**

No	Skenario Pengujian	Test Case	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
1	Mengonfirmasi bahwa Plaintext, lalu langsung mengklik tombol enkripsi	Plaintext: Kosong	Sistem akan menolak mengenkripsi dan menampilkan pesan "Plaintext harus diisi !!!"	Sesuai harapan	Valid
2	Mengonfirmasi bahwa Chiphertext, lalu langsung mengklik tombol dekripsi	Chiphertext: Kosong	Sistem akan menolak mendekripsi dan menampilkan pesan "Chiphertext harus diisi !!!"	Sesuai harapan	Valid
3	Mengonfirmasi bahwa Plaintext, lalu langsung mengklik tombol enkripsi	Plaintext: Ada	Sistem akan menampilkan hasil enkripsi	Sesuai harapan	Valid
4	Mengonfirmasi bahwa Chiphertext, lalu langsung mengklik tombol dekripsi	Chiphertext: Ada	Sistem akan menampilkan hasil dekripsi	Sesuai harapan	Valid

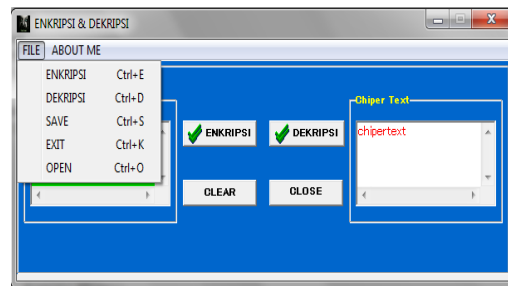
Sumber: Pribadi (hasil penelitian tahun 2013)

**Implementasi**



Gambar :4.3. Implementasi enkripsi

Sumber : Pribadi (hasil penelitian tahun 2013)



Gambar : 4.4. Implementasi dekripsi

Sumber:Pribadi (hasil penelitian tahun 2013)

**KESIMPULAN**

a. Kesimpulan

Berdasarkan hasil penelitian dan pengujian yang telah dilakukan dapat disimpulkan :

1. Pemrograman berorientasi objek terbukti dapat digunakan dalam membangun aplikasi enkripsi dan dekripsi
2. Dalam pengujian aplikasi tombol enkripsi dan dekripsi sudah dapat diterapkan dengan baik sesuai dengan algoritma yang digunakan
3. Algoritma Julius Caesar lebih mudah dan sederhana dalam melakukan enkripsi dan dekripsi pesan teks, dengan cara menggunakan kombinasi kunci yang diinginkan

- b. Saran
1. Aplikasi ini hanya dapat dijalankan pada system operasi windows,
  2. Dimasa yang akan datang diharapkan ada pengembangan aplikasi yang dapat diterapkan pada sistem operasi keluarga linux
  3. Pengembangan aplikasi dimasa yang akan datang juga di harapkan dapat berbasis mobile.

### UCAPAN TERIMA KASIH

Pada kesempatan yang berbahagia ini, ijinkanlah kami penulis ingin mengucapkan terimakasih kepada :

1. Ketua STMIK Nusamandiri
2. Ketua Jurusan Program Studi Teknik Informatika
3. Kepada semua rekan deosen di lingkungan STMIK Nusamandiri
4. Seluruh Mahasiswa Teknik Informatika STMIK Nusamandiri
5. Dan kepada semua pihak yang tidak dapat kami sebutkan nama satu persatu

### DAFTAR PUSTAKA

- Ariyus, Doni, 2006 Kriptografi Keamanan Data dan Komunikasi, Graha Ilmu Yogyakarta,
- Bahri, Saiful, Kusnassriyanto dan Wawan Sjachriyanto, 2005, Informatika, Bandung
- Binanto, Iwan, 2005, Konsep Bahasa Pemrograman, Andi Offset, Yogyakarta
- Cahyadi, Tri, 2012, Implementasi Steganografi LSB Dengan Enkripsi Vigenere Cipher Pada Citra JPEG, ISSN : 2302-9927, 282 Semarang No 4 Desember 2012 1-8 <http://ejournal-s1.undip.ac.id>
- Latief, Mukhlisufatih, 2010, Studi Perbandingan Enkripsi Menggunakan IDEA dan MMB Gorontalo: Media Elektrik Vol 5 <http://elektro.unm.ac.id/jurnal/ME.%2>

Nugroho, Adi. 2010. Rekayasa Perangkat Lunak Berorientasi Objek dengan metode USDP. Yogyakarta: ANDI.

Pressman, Roger. 2010. Rekayasa Perangkat Lunak. Yogyakarta: Andi Yogyakarta

Sholiq. 2006. Pemodelan Sistem Informasi Berorientasi Objek. Yogyakarta: Graha Ilmu.

Sudrajat, Wahyu, Antonius. 2006. Implementasi Enkripsi Database Menggunakan Transparent Data Encryption Pada Database Engine Oracle. Palembang: Jurnal Ilmiah STMIK GI MDP Vol.2, No.3 Oktober 2006: 14-19. Diambil dari: <http://eprints.mdp.ac.id/544/1/Jurnal%20Implementasi%20Enkripsi%20Database.pdf> (29 April 2013)

Yulikuspartono. 2004. Pengantar Logika dan Algoritma. Yogyakarta: Andi Yogyakarta