# PERANCANGAN SISTEM PENERAPAN KRIFTOGRAFI KEAMANAN DATA SISWA DI SMK TR PANCA BUDI 1 MEDAN MENGGUNAKAN METODE VIGENERE CHIPER

Disusun dan Diajukan Sebagai Salah Satu Syarat untuk Menempuh Ujian Akhir
Memperolah Gelar Sarjana Komputer Pada Fakultas Sains Dan Teknologi
Universitas Pembangunan Panca Budi Medan

**SKRIPSI**

**OLEH**


| | |
|---|---|
| NAMA | : REGITA AFRILLA |
| N.P.M | : 1514370372 |
| PROGRAM STUDI | : SISTEM KOMPUTER |


**FAKULTAS SAINS DAN TEKNOLOGI**
**UNIVERSITAS PEMBANGUNAN PANCA BUDI**
**MEDAN**
**2020**

# ABSTRAK

**REGITA AFRILLA**

**Perancangan Sistem Penerapan Kriptografi Keamanan Data Siswa Di SMK TR Panca Budi 1 Medan Menggunakan Metode Vigenere Chiper**

**2020**

Kriptografi merupakan salah satu metode mengamankan data yang dapat digunakan untuk menjaga kerahasiaan data, keaslian data sera keaslian pengirim. Metode ini bertujuan agar informasi yang bersifat rahasia yang dikirm melalui telekomunikasi umum seperti LAN atau Internet. Kriptografi biasanya dalam bentuk enkripsi dan Deskripsi. Untuk menyembunyikan tulisan, biasanya mengguakan algoritma. Algoritma yang dipakai dalam aplikasi ini adalah Algoritma Vigenere Cipher. Dalam hal ini, penulis berkeinginan mengangkat topik enkripsi dan deskripsi menjadi sebuah penulisan ilmiah skripsi dengan menggunakan visual studio yang berkembang saat ini. Diharapkan dengan adanya aplikasi ini, mahasiswa serta dosen dapat melakukan uji coba enkripsi menggunakan algoritma Vigenere Cipher.

**Kata Kunci**: Kriptografi, *Vigenere Cipher*.

# DAFTAR ISI

**Halaman**

# BAB I

# PENDAHULUAN

## 1.1. Latar Belakang Masalah

Perkembangan teknologi komputer dan jaringan komputer yang semakin pesat, tidak hanya memberikan dampak positif seperti kemudahan dan kepraktisan dalam mengolah informasi dan data, namun juga dapat memberikan dampak negatif seperti penyalahgunaan informasi dan data, hal ini dikarenakan manusia yang selalu bereksperimen dalam mengembangkan teknologi komputer dan juga mencari celah atau kelemahan pada sistem komputer. Data dan informasi tidak cukup pengembangannya hanya difokuskan pada kemudahan dan kepraktisan saja dalam mengolah dan mengaksesnya, namun dibutuhkan juga sistem pengamanan yang memadai dan terjamin, salah satu cara yang paling baik digunakan untuk mengamankan data menggunakan metode kriptografi. (Pratama, 2015)

SMK TR Panca Budi 1 Medan adalah salah satu sekolah yang ada di Indonesia yang tempatnya berada di Kota Medan, Sumatera Utara. Dalam sejarah berdirinya Perguruan Panca Budi sejak tahun 1961 sampai sekarang ini, Panca Budi berkembang secara alami dan bersifat sosial, karena Yayasan pada waktu itu berfokus membina kegiatan-kegiatan keagamaan. Pada tahun 1997 unit SMK TR (Teknologi Rekayasa) didirikan dan mulai berkembang sampai saat ini. SMK TR saat ini banyak menggunakan sistem teknologi untuk menjalankan suatu program atau untuk menginput suatu data sekolah, saat ini keamanan dalam menginput data siswa sangatlah dibutuhkan.

Dalam sistem penyimpanan data pada SMK TR Panca Budi 1 Medan masih menggunakan cara penyimpanan yang manual. Penyimpanan data yang digunakan pada SMK TR Panca Budi 1 Medan ini adalah masih menggunakan penyimpanan dengan *microsoft excel* sebagai pendataan siswanya. Data siswa yang terdapat pada file *excel* tersebut masih dalam bentuk *plaintext* yang dapat dibaca oleh orang lain dan file juga rentan hilang maupun terkena virus. Dilihat dari minimnya keamanan data siswa yang ada pada SMK TR Panca Budi 1 Medan ini membuat penulis ingin membuat suatu program aplikasi yang dapat menyimpan data siswa kedalam suatu database dan memberikan enkripsi pada data yang tersimpan tersebut dengan menggunakan metode enkripsi *Vigenere Chiper.* Sistem yang akan dibuat nantinya berupa aplikasi yang berbasis desktop untuk melakukan penginputan data dan memberikan enkripsi pada data tersebut. Aplikasi yang akan dibuat nantinya bertujuan untuk memudahkan pegawai dalam menginputkan data siswa dan memberikan keamanan lebih pada data siswa yang telah diinputkan. Sehingga dapat lebih membantu pihak sekolah SMK TR Panca Budi 1 Medan dalam mengamankan data siswanya agar tidak terjadi kebocoran data pribadi yang berdampak pada data siswa yang bocor tersebut.

Berdasarkan latar belakang diatas maka penulis tertarik untuk memilih judul "**Perancangan Sistem Penerapan Kriptografi Keamanan Data Siswa Di SMK TR Panca Budi 1 Medan Menggunakan Metode *Vigenere Chiper*".**

## 1.2. Rumusan Masalah

Adapun permasalahan yang dihadapi dalam merancang sistem penerapan kriptografi keamanan data siswa di SMK TR Panca Budi 1 Medan ini adalah :

1) Bagaimana membuat keamanan data siswa pada SMK TR Panca Budi 1 medan agar tidak dapat dibaca oleh sembarang orang?
2) Bagaimana membuat sistem yang dapat lebih memudahkan sekolah SMK TR Panca Budi dalam melakukan input data siswanya?

## 1.3. Batasan Masalah

Dalam merancang sistem penerapan kriptografi keamanan data siswa di SMK TR Panca Budi 1 Medan ini, penulis membatasi masalah sebagai berikut :

1) Aplikasi yang dibuat hanya pada pendataan siswa dan mengamankan data dengan menggunakan metode *vigenere chiper*.
2) Sistem yang dirancang nantinya dibuat dengan menggunakan bahasa pemrograman visual basic yang berbasis desktop dan menggunakan *database MySQL*.
3) Hanya menggunakan data siswa di SMK TR Panca Budi dengan mengikuti format di SMK tersebut.
4) Ruang lingkup pengamanan data tidak mencakup data guru, jadwal mata pelajaran dan kegiatan siswa di SMK TR Panca Budi.
5) Data yang di enkripsi berupa data text, simbol, tanda baca dan spasi yang ada didalam file tersebut.

6) Jumlah karakter yang digunakan dalam 1 file adalah yang sesuai dengan tabel asci (255 karakter dan simbol)
7) Kapasitas file maksimal 2 MB dengan format .doc atau .docx.

## 1.4. Tujuan Penelitian

Tujuan yang ingin dicapai penulis dalam merancang sistem penerapan kriptografi keamanan data siswa di SMK TR Panca Budi 1 Medan ini adalah :

1) Agar dapat memaksimalkan keamanan data yang ada di SMK TR Panca Budi 1 Medan terkhususnya data siswa.
2) Agar memiliki sebuah pangkalan data untuk menyimpan data siswa kedalam sistem yang sudah terkomputerisasi.

## 1.5. Manfaat Penelitian

Merancang sistem penerapan kriptografi keamanan data siswa di SMK TR Panca Budi 1 Medan ini antara lain:

1) Memiliki sebuah *database* untuk menyimpan data siswa pada sebuah sistem tanpa memakan banyak tempat untuk menyimpan arsip berkasnya.
2) Data siswa yang terenkripsi tersebut tidak dapat dibaca oleh orang lain yang tidak memiliki hak akses karena data yang terenkripsi dibuat secara acak.

# BAB II

# LANDASAN TEORI

## 2.1. Kemanan Data

Pada zaman teknologi informasi sekarang, data atau informasi merupakan suatu asset yang sangat berharga dan harus dilindungi. Hal ini juga diikuti oleh kemajuan teknologi komputer. Kemajuan teknologi komputer membantu semua aspek kehidupan manusia. Dengan adanya kemajuan dalam teknologi informasi, komunikasi dan komputer maka kemudian muncul masalah baru, yaitu masalah keamanan akan data dan informasi dan dalam hal ini akan membuka peluang bagi orang-orang yang tidak bertanggung jawab untuk menggunakannya sebagai tindak kejahatan. Dan tentunya akan merugikan pihak tertentu. Dalam keamanan data ada beberapa aspek yang berkaitan dengan persyaratan kemanan yaitu(**Pabokory, 2015**):

1. *Secrecy*. Berhubungan dengan akses membaca data dan informasi. Data dan informasi di dalam suatu sistem komputer hanya dapat diakses dan dibaca oleh orang yang berhak.
2. *Integrity*. Berhubungan dengan akses merubah data dan informasi. Data dan informasi di dalam suatu sistem komputer hanya dapat diubah oleh orang yang berhak.

3. *Availability*. Berhubungan dengan ketersediaan data dan informasi. Data dan informasi yang berada dalam suatu sistem komputer tersedia dan dapat dimanfaatkan oleh orang yang berhak. **(Pabokory, 2015).**
4. Lebih lanjut menurut **(Pabokory, 2015)**, terdapat lima langkah kemanan komputer yang baik untuk diperhitungkan yaitu; aset, analisis resiko, perlindungan, alat dan prioritas.

## 2.2. *Kriptografi*

*Kriptografi* merupakan kata dari bahasa Yunani yaitu cryptography, terdiri dari dua suku kata yaitu kripto dan graphia. Kripto artinya menyembunyikan, sedangkan graphia artinya tulisan. Sehingga, bila digabungkan akan menjadi kata yang berarti menyembunyikan/merahasiakan tulisan. *Kriptografi* adalah suatu ilmu ataupun seni mengamankan pesan dan dilakukan oleh *cryptographer* (**Anonim, 2014**).

Menurut (**Rhee, 2013**).*kriptografi* digunakan untuk memastikan privasi dan autentifikasi data dalam komunikasi antar sistem komputer. Terdapat dua proses dasar dalam *kriptografi* yaitu:

1. *Enkripsi*, adalah sebuah proses yang melakukan perubahan sebuah kode dari yang bisa dimengerti menjadi sebuah kode yang tidak bisa dimengerti (tidak terbaca). (**Pabokory, 2015**)

2. *Deskripsi*, adalah kebalikan dari *Enkripsi* yaitu mengubah kembali bentuk tersamar tersebut menjadi informasi awal. (**Pabokory, 2015**).

Sebuah pesan atau data yang masih asli dan belum mengalami penyandian dikenal dengan istilah plaintext. Kemudian setelah disamarkan dengan suatu cara penyandian, maka plaintextini disebut sebagai chipertext. Proses penyamaran dari plaintextke *cipher*text disebut *Enkripsi* (encryption), dan proses pengembalian dari *cipher*textmenjadi plaintext kembali disebut dekripsi (decryption). (**Pabokory, 2015**).*File* yang dapat di*Enkripsi* dapat berupa teks, gambar maupun audio dan video.

## 2.3. *Kriptografi Vigenere Cipher*

Kriptografi*Vigenere Cipher* merupakan bagian dari kriptografi polialfabetik yang ditemukan pertama kali pada tahun 1586 oleh diplomat Perancis bernama Blaise de Vigenere (1523-1596). Menurut (**Prade E,2014**) *Vigenere Cipher* merupakan jenis *cipher* abjad majemuk yang paling sederhana. *Vigenere Cipher* menerapkan metode substitusi poli alfabetik dan termasuk kedalam kategori kunci simetris dimana kunci yang digunakan untuk proses enkripsi adalah sama dengan kunci yang digunakan untuk proses dekripsi. Tujuan utama dari *vigenere cipher* ini adalah menyembunyikan keterhubungan antara *plaintext* dan *ciphertext* dengan menggunakan kata kunci sebagai penentu pergeseran karakternya.

Tabel yang digunakan merupakan tabel 26 huruf alfabetik standart, yang dimulai dari A sampai Z. Panjang kunci tersebut bisa lebih pendek ataupun sama

dengan *plaintext*, maka kunci tersebut akan diulang secara periodik hingga panjang kunci tersebut sama dengan panjang *plaintext*nya. Berikut ini rumus enkripsi dan dekripsi *Vigenere Cipher* :

Enkripsi : Ci = Pi + ki mod 26

Dekripsi : Pi = Ci – ki mod 26

Ci : *Ciphertext*

Pi : *Plaintext*

ki : *Key* atau kunci

Menurut (**Teady, 2015**) dalam jurnalnya yang berjudul Vigenere Cipher Menggunakan Spreadsheet, penyandian Vigenere atau Vigenere Cipher merupakan salah satu teknik penyandian dengan cara subtitusi. Bruen (**Bruen, 2015**) dalam bukunya Cryptography, Information Theory, and Error-Correction, serta Martin (**Martin 2015**) dalam bukunya Everyday Cryptography, mengatakan bahwa vigenere cipher adalah sebuah metode dari enkripsi teks alfabetik menggunakan serangkaian penyandian bebasis caesar pada huruf-huruf dari sebuah kata kunci, dan merupakan bentuk sederhana dari subtitusi polyalphabetic.

Teknik subtitusinya serupa dengan semua penyandian berbasis caesar. Seperti penyandian berbasis caesar lainnya, vigenere cipher sebenarnya juga melakukan pergeseran, tetapi pergeseran dilakukan perhuruf dengan huruf berikutnya pada plaintext berbeda. Dengan demikian jika pada caesar cipher seseorang dengan mudah menebak kuncinya dengan melakukan pergeseran abjad mulai dari 1 s/d 26 secara cara try and error, sampai ditemukan nilai kunci yang tepat. Maka pada pada vigenere cipher akan lebih sulit menebak kuncinya dengan

try and error mencari nilai pergeseran seperti pada Caesar cipher, karena antara huruf yang satu dengan huruf berikutnya mempunyai nilai pergeseran yang berbeda.

Berbeda dengan caesar cipher, input key vigenere cipher berupa sebuah kata yang merupakan rangkaian huruf.

Contoh :

Input Kunci : cerdas

Pengulangan : cerdascerdascerdasc

Plainteks : belajarsupayapandai

cipherteks : dicdjstwlsaqctrqdsk

Key akan diulang-ulang sampai sejumlah huruf pada plaintext.

Kemudian setiap huruf satu persatu akan dicari pada table vigenere sesuai dengan keynya.

Dalam jurnalnya yang berjudul Implementasi Enkripsi Data Dengan Algoritma Vigenere Cipher. Vigenere Chiper termasuk dalam cipher abjadmajemuk (polyalphabetic substitution Chiper) yang dipublikasikan oleh diplomat (sekaligus seorang kriptologis) Perancis, Blaise de Vigenere pada abad 16 (tahun 1586).

*Vigenere Cipher* menggunakan suatu kunci yang memiliki panjang tertentu. Panjang kunci tersebut bias lebih pendek ataupun sama dengan plainteksnya. Jika panjang kunci kurang dari panjang plainteks, maka kunci tersebut akan diulang secara periodic sehingga panjang kunci tersebut akan sama panjang dengan plainteksnya.

Formula atau rumus enkripsi *Vigenere Cipher*

Ci = (Pi + Ki) mod 26

Formula atau rumus deskripsi *Vigenere Cipher*

Pi = (Ci-Ki) mod 26 ; untuk Ci >= Ki

Pi = (Ci+26-Ki) mod 26 ; untuk Ci <= Ki

Dengan penjelasan:

Ci = nilai decimal karakter *ciphertext* ke-i

Pi = nilai decimal karakter *ciphertext* ke-i

Ki = nilai decimal karakter *ciphertext* ke-i

Contoh bila kita ingin memberi sandi pada kata ILMU KOMPUTER dank unci yang kita inginkan adalah UNPAB maka enkripsi dilakukan sebagai berikut:

Plainteks : ILMUKOMPUTER

Kunci : UNPAB

Karena metode yang diterapkan pada *Vigenere Cipher* adalah dengan menyusun kunci yang panjangnya akan disesuaikan dengan panjang plainteksnya, maka kunci akan mengalami perulangan sampai memenuhi banyak karakter yang terdapat pada plainteks.

Plainteks : ILMUKOMPUTER

Kunci : UNPABUNPABUN

Langkah selanjutnya adalah masuk ke proses enkripsi dengan metode vigenere menggunakan formula atau rumus Ci = (Pi + Ki) mod 26.

Plainteks : I L M U K O M P U T E R

Kunci : U N P A B U N P A B U N

Cipherteks :B Y B U L I Z E U U Y E

Cipherteks didapatkan dengan cara menggunakan formula Ci = (Pi + Ki) mod 26

C1 = (I + U) mod 26 (8+20) mod 26 = 2 = B

C2 = (L + N) mod 26 (11+13) mod 26 = 24 =Y

Setelah seluruh karakter mendapatkan cipherteks, maka proses enkripsi sudah selesai.

Proses deskripsi dilakukan untuk memecahkan cipherteks kembali menjadi plainteks. Untuk proses deskripsi digunakan rumus atau formula Pi = (Ci-Ki) mod 26 ; untuk Ci >= Ki dan/atau Pi = (Ci+26-Ki) mod 26 ; untuk Ci <= Ki

Cipherteks :B Y B U L I Z E U U Y E

Kunci : U N P A B U N P A B U N

Plainteks : I L M U K O M P U T E R

Seperti pada karakter pertama dimana C1 bernilai 2 dan K1 bernilai 20 maka digunakan formula atau rumus Pi = (Ci+26-Ki) mod 26 karena nilai C1 <= K1

P1 = (2+26-20) mod 26 = 8 = I

Sementara pada karakter kedua dimana C2 bernilai 24 dan K2 bernilai 13 maka digunakan formula atau rumus Pi = (Ci-Ki) mod 26 karena nilai C2 >= K2

P2 = (24-13) mod 26 = 11 = L

Selain itu kunci juga dapat berupa angka, berikut adalah table huruf beserta angka yang akan membantu proses enkripsi.

**Tabel 2.1 Tabel Alphabet dan Konversinya**

| 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: Arjana, 2013)**

Misalkan plainteksnya adalah ILMUKOMPUTER dan kunci yang ingin diberikan adalah 5,15,8,0 maka selanjutnya didapatkan tabel enkripsi sebagai berikut.

**Tabel 2.2 Tabel Enkripsi**

| I | L | M | U | K | O | M | P | U | T | E | R |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 8 | 11 | 12 | 20 | 10 | 14 | 12 | 15 | 20 | 19 | 4 | 17 |
| 5 | 15 | 8 | 0 | 5 | 15 | 8 | 0 | 5 | 15 | 8 | 0 |

*Ciphertext* yang dihasilkan adalah sebagai berikut:

*Ciphertext* : 13,1,20,20,15,29,20,15,25,9,12,17

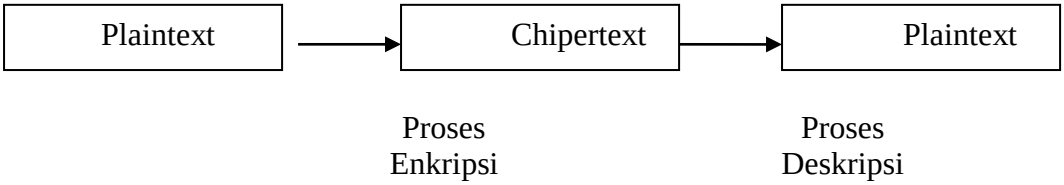
Untuk proses deskripsi dapat dilakukan dengan rumus atau formula yang sama seperti *vigenere cipher* dengan menggunakan huruf . Misalkan untuk angka pertama chipertext adalah 13 dan mau didapatkan plainteksnya maka;

P1 = (13-5) modulo 26 = 8 = I

## 2.4. *Enkripsi*

*Enkripsi* merupakan hal yang sangat penting dalam *kriptografi* supaya keamanan data yang dikrimkan bisa terjaga kerahasiaannya. Pesan asli (plaintext)

diubah menjadi kode-kode yang tidak dimengerti. *Enkripsi* bisa diartikan dengan chiper atau kode. Sama halnya dengan kita yang tidak mengerti sebuah kata, kita akan dapat melihatnya di dalam kamus atau daftar istilah-istilah. Berbeda halnya dengan *Enkripsi*, untuk mengubah plaintext ke bentuk chipertext, kita harus menggunakan algoritma yang dapat mengkodekan data yang kita inginkan. Berikut adalah penggambaran proses *Enkripsi*.




**Gambar 2.1. Proses *Enkripsi* dan *Deskripsi***

(**Sumber: Pabokory, 2015)**

## 2.5. *Kriptografi* Klasik

Menurut (Bishop, 2014).*kriptografi* klasik adalah *kriptografi* yang disebut juga sebagai *kriptografi* kunci tunggal atau *kriptografi* simetris yang menggunakan kunci yang sama untuk *Enkripsi* maupun *Deskripsi*. *Kriptografi* klasik merupakan *kriptografi* yang digunakan pada zaman dahulu sebelum komputer ditemukan atau sudah ditemukan namun belum secanggih sekarang. *Kriptografi* ini melakukan pengacakan huruf pada kata terang / plaintext.

## 2.6. *One Time Pad (OTP)*

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 hasi XOR antara *bitplaintext*dan *bitkey*. Pada metode ini *plain text* diubah kedalam kode ASCII dan kemudian dikenakan operasi XOR terhadap kunci yang sudah diubah ke dalam kode ASCII. (**Hamokwarong,2014**).

*One-time pad* adalah salah satu stream *Cipher* klasik yang secara matematis terbukti sempurna aman. *Cipher* teksnya 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, 2014**)

Sayangnya *one-time pad* tidak dapat diimplementasikan 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, 2014**)

## 2.7. Algoritma

Penyelesaian permasalahan dengan menggunakan alat bantu system computer paling tidak akan melibatkan lima tahapan, yaitu:

1. Analisis masalah
2. Merancang algoritma
3. Membuat program computer
4. Menguji hasil program computer
5. Dokumentasi

Poin kedua menerangkan bahwa dalam perancangan sebuah system computer dibutuhkan adanya perancangan algoritma. Sehingga setelahnya dapat dilanjutkan ke tahap-tahap berikutnya hingga dokumentasi.

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 softwarekomputer. Software terbangun atas susunan program (silahkan baca mengenai pengertian program) dan syntax (cara penulisan/pembuatan program). Untuk menyusun program atau syntax, diperlukannya langkah-langkah 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.

Pengertian Algoritma adalah susunan yang logis dan sistematis untuk memecahkan suatu masalah atau untuk mencapai tujuan tertentu.

Algoritma adalah kunci dari bidang ilmu komputer, dan pada dasarnya setiap hari kita melakukan aktivitas algoritma. Kata algoritma berasal dari sebutan Algorizm (Abu Abdullah Muhammad Ibn Musa Al Khwarizmi, ahli matematika Uzbeki

a. Algoritma adalah urutan langkah-langkah berhingga untuk memecahkan masalah logika atau matematika
b. Algoritma adalah logika, metode dan tahapan (urutan) sistematis yang digunakan untuk memecahkan suatu permasalahan.
c. Algoritma adalah urutan langkah-langkah logis penyelesaian masalah yang disusun secara sistematis dan logis.
d. Algoritma adalah urutan logis pengambilan keputusan untuk pemecahan masalah.

Pembuatan algoritma harus selalu dikaitkan dengan:

a. Kebenaran algoritma
b. Kompleksitas (lama dan jumlah waktu proses dan penggunaan memori)

Kriteria Algoritma yang baik:

a. Tepat, benar, sederhana, standar dan efektif
b. Logis, terstruktur dan sistematis

c. Semua operasi terdefinisi

d. Semua proses harus berakhir setelah sejumlah langkah dilakukan

e. Ditulis dengan bahasa yang standar dengan format pemrograman agar mudah untuk diimplementasikan dan tidak menimbulkan arti ganda.

## 2.8. *Unified Modeling Language* (UML)

### 2.8.1 Pengenalan UML

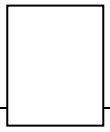
*Unified Modelling Language* (UML) adalah suatu alat untuk memvisualisasikan dan mendokumentasikan hasil analisis dan desain yang berisi sintak dalam memodelkan sistem secara visual (**Haviluddin, 2015**). Banyak orang yang telah membuat bahasa pemodelan pembangunan perangkat lunak sesuai dengan teknologi pemrograman yang berkembang pada saat itu, misalnya yang sempat berkembang dan digunakan oleh banyak pihak adalah *DataFlow Diagram* (DFD) untuk memodelkan perangkat lunak yang menggunakan pemrograman prosedural atau struktur, kemudian juga ada *State Transition Diagram* (STD) yang digunakan untuk memodelkan *real time* (waktu nyata).

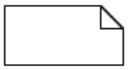
Pada perkembangan teknik pemrograman berorientasi objek, muncullah sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemrograman berorientasi objek, yaitu *Unified Modeling Language* (UML).

### 2.8.2. *Use Case Diagram*

Diagram yang menggambarkan *actor*, *use case* dan relasinya sebagai suatu urutan tindakan yang memberikan nilai terukur untuk aktor. Sebuah *use case* digambarkan sebagai elips horizontal dalam suatu diagram *use case diagram* (**Haviluddin, 2015**).

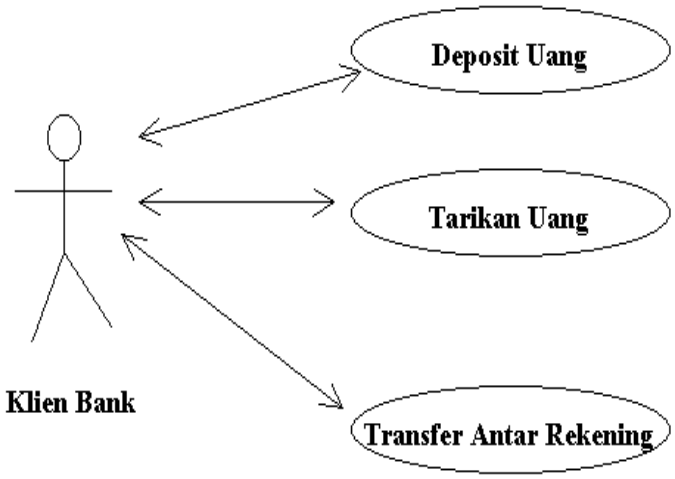
**Tabel 2.3 Simbol *Use Case Diagram***

| NO | GAMBAR | NAMA | KETERANGAN |
|---|---|---|---|
| 1 |  | *Actor* | Menspesifikasikan himpuan peran yang pengguna mainkan ketika berinteraksi 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, 2015**)

Contoh Use Case Diagram :




**Gambar 2.2. Contoh *Use Case Diagram***
Sumber : (**Haviluddin, 2015)**

### 2.8.3. *Activity Diagram*

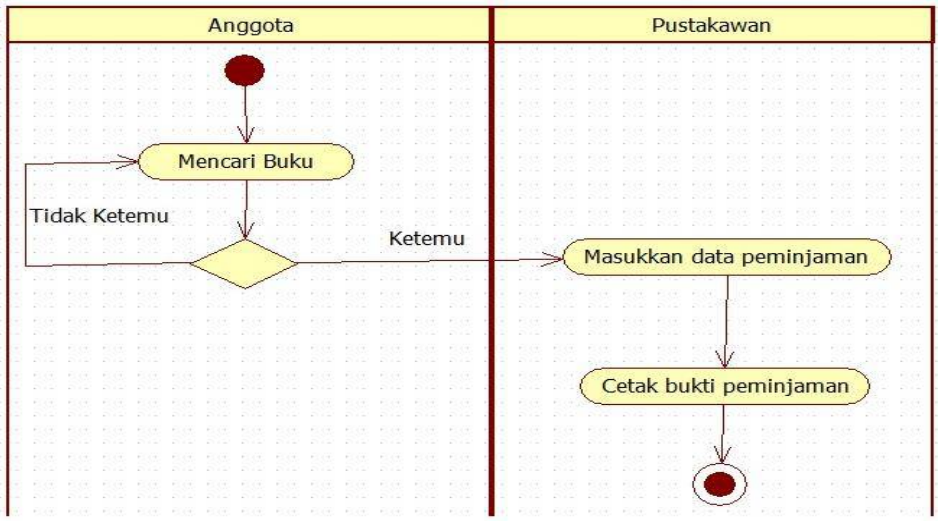
Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau *menu* yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor,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

**Tabel 2.4 Simbol *Activity Diagram***

| NO | GAMBAR | NAMA | KETERANGAN |
|---|---|---|---|
| **1** |  | *Activity* | Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi 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 :



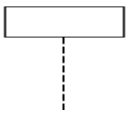

**Gambar 2.3. Contoh *Activity Diagram***

Sumber : (Gellysa Urva, 94 : 2015)

### 2.8.4. *Sequence Diagram*

Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan 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 diinstansiasi menjadi objek itu. Membuat diagram sekuen juga dibutuhkan untuk melihat skenario yang ada pada *use case*.

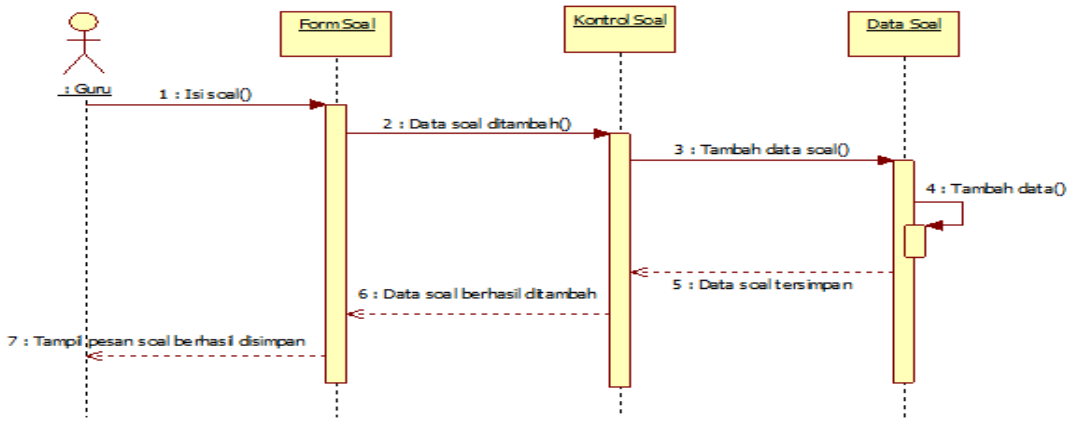
**Tabel 2.5 Simbol *Sequence Diagram***

| 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 :



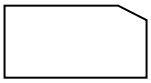

**Gambar 2.4. Contoh *Squence Diagram***
Sumber : (Gellysa Urva, 95 : 2015)

### 2.8.5. *Class Diagram*

*Class diagram* menggambarkan struktur statis dari kelas dalam sistem anda dan menggambarkan atribut, operasi dan hubungan antara kelas. Class diagram membantu dalam memvisualisasikan struktur kelas-kelas dari suatu sistem dan merupakan tipe diagram yang paling banyak dipakai. Selama tahap

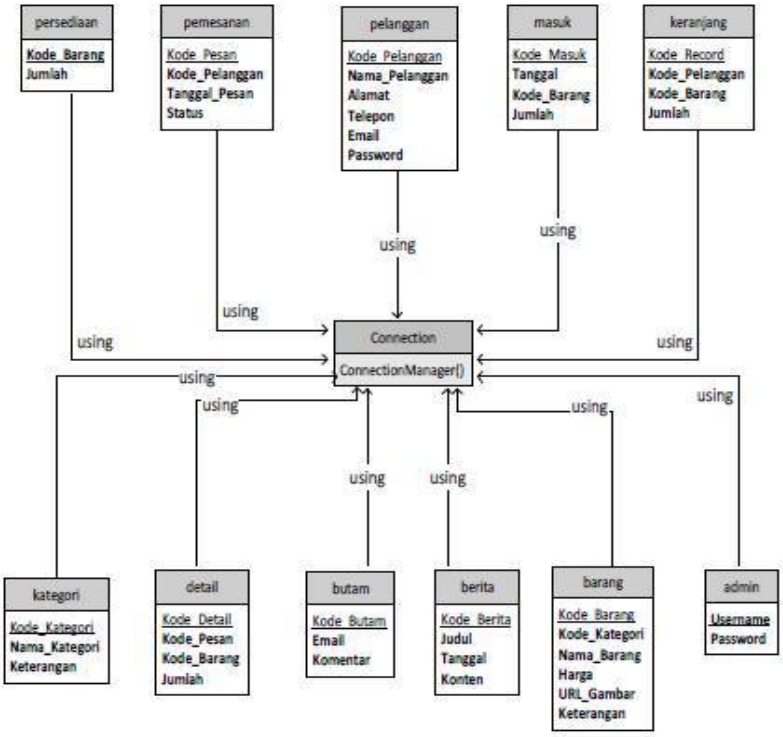
desain, class diagram berperan dalam menangkap struktur dari semua kelas yang membentuk arsitektur sistem yang dibuat.

**Tabel 2.6 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* | Menspesifikasikan bahwa use case target memperluas prilaku dari use case sumber pada suatu titik yang diberikan. |

Sumber : (Gellysa Urva, 95 : 2015)

Contoh *Class* Diagram :




**Gambar 2.5. Contoh *Class Diagram***

Sumber : (Gellysa Urva, 95 : 2015)

## 2.9. 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, 2013**).

Informasi bisa menjadi fungsi penting dalam membantu mengurangi rasa cemas pada seseorang. Menurut pendapat (**Notoatmodjo, 2018)** 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.

Adapun fungsi-fungsi informasi adalah sebagai berikut:

1. Untuk meningkatkan pengetahuan bagi si pemakai.
2. Untuk mengurangi ketidakpastian dalam proses pengambilan keputusan pemakai.
3. 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.

## 2.10. Pengertian Visual Studio

*Visual Studio .Net* merupakan salah satu *tool development Microsoft* yang dapat digunakan untuk membuat aplikasi di lingkungan kerja berbasis sistem operasi *Windows*. *Visual Studio .NET* menyediakan tools bagi para *developer* untuk membangun aplikasi yang berjalan di .*Net Framework* (**Safik, 2015**).

*Visual Studio* (*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 Studio* merupakan turunan bahasa pemrograman *STUDIO* 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 Studio* (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 Studio* merupakan turunan bahasa pemrograman *STUDIO* dan menawarkan pengembangan perangkat lunak komputer berbasis grafik dengan cepat, Beberapa bahasa skrip seperti *Visual Studio for Applications* (*VBA*) dan

*Visual Studio Scripting Edition* (*VBScript*), mirip seperti halnya *Visual Studio*, tetapi cara kerjanya yang berbeda.

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

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

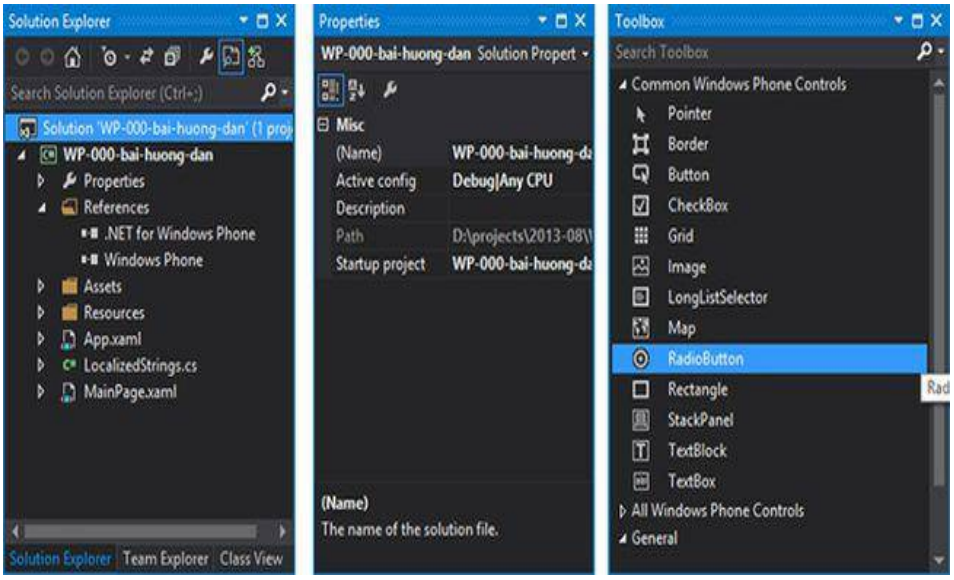
### 2.10.1 Komponen kerja

Beberapa komponen kerja program *visual Studio 2015* 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 Studio 2015 sesuai dengan yang kita butuhkan. Berikut ini adalah beberapa komponen kerja dari visual Studio 2015 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 Studio 2015*

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




**Gambar 2.6. Tampilan *Toolbox***
Sumber : (**Safik, 2015**).

**Tabel 2.7*Toolbox Visual Studio***

| Nama tombol | fungsi |
|---|---|
| *Pointer* | Memilih, mengatur ukuran dan memindahkan posisi yang terpasang di bagian form. |
| *Bindingsources* | Untuk mengkoneksikan 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 *bit*map, 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, 2015**).

## 2.11. Tabel ASCII

ASCII merupakan kepanjangan dari (American Standard Code for Information Interchange), dan pengertian dari ASCII sendiri adalah suatu standar internasional dalam kode huruf dan simbol seperti Hex dan Unicode tetapi ASCII lebih bersifat universal, contohnya 124 adalah untuk karakter "|". Ia selalu digunakan oleh komputer dan alat komunikasi lain untuk menunjukkan teks. sedangkan fungsi dari kode ASCII ialah digunakan untuk mewakili karakter-karakter angka maupun huruf didalam komputer, sebagai contoh dapat kita lihat pada karakter 1, 2, 3, A, B, C, dan sebagainya.

| DEC | OCT | HEX | BIN | Symbol |
|---|---|---|---|---|
| 0 | 000 | 00 | 00000000 | NUL |
| 1 | 001 | 01 | 00000001 | SOH |
| 2 | 002 | 02 | 00000010 | STX |
| 3 | 003 | 03 | 00000011 | ETX |
| 4 | 004 | 04 | 00000100 | EOT |
| 5 | 005 | 05 | 00000101 | ENQ |
| 6 | 006 | 06 | 00000110 | ACK |
| 7 | 007 | 07 | 00000111 | BEL |
| 8 | 010 | 08 | 00001000 | BS |
| 9 | 011 | 09 | 00001001 | HT |
| 10 | 012 | 0A | 00001010 | LF |
| 11 | 013 | 0B | 00001011 | VT |
| 12 | 014 | 0C | 00001100 | FF |
| 13 | 015 | 0D | 00001101 | CR |
| 14 | 016 | 0E | 00001110 | SO |
| 15 | 017 | 0F | 00001111 | SI |

| | | | | |
|---|---|---|---|---|
| 16 | 020 | 10 | 00010000 | DLE |
| 17 | 021 | 11 | 00010001 | DC1 |
| 18 | 022 | 12 | 00010010 | DC2 |
| 19 | 023 | 13 | 00010011 | DC3 |
| 20 | 024 | 14 | 00010100 | DC4 |
| 21 | 025 | 15 | 00010101 | NAK |
| 22 | 026 | 16 | 00010110 | SYN |
| 23 | 027 | 17 | 00010111 | ETB |
| 24 | 030 | 18 | 00011000 | CAN |
| 25 | 031 | 19 | 00011001 | EM |
| 26 | 032 | 1A | 00011010 | SUB |
| 27 | 033 | 1B | 00011011 | ESC |
| 28 | 034 | 1C | 00011100 | FS |
| 29 | 035 | 1D | 00011101 | GS |
| 30 | 036 | 1E | 00011110 | RS |
| 31 | 037 | 1F | 00011111 | US |
| DEC | OCT | HEX | BIN | Symbol |
| 32 | 040 | 20 | 00100000 | |
| 33 | 041 | 21 | 00100001 | ! |
| 34 | 042 | 22 | 00100010 | " |
| 35 | 043 | 23 | 00100011 | # |
| 36 | 044 | 24 | 00100100 | $ |
| 37 | 045 | 25 | 00100101 | % |
| 38 | 046 | 26 | 00100110 | & |
| 39 | 047 | 27 | 00100111 | ' |
| 40 | 050 | 28 | 00101000 | ( |
| 41 | 051 | 29 | 00101001 | ) |
| 42 | 052 | 2A | 00101010 | * |
| 43 | 053 | 2B | 00101011 | + |
| 44 | 054 | 2C | 00101100 | , |
| 45 | 055 | 2D | 00101101 | - |

| | | | | |
|---|---|---|---|---|
| 46 | 056 | 2E | 00101110 | . |
| 47 | 057 | 2F | 00101111 | / |
| 48 | 060 | 30 | 00110000 | 0 |
| 49 | 061 | 31 | 00110001 | 1 |
| 50 | 062 | 32 | 00110010 | 2 |
| 51 | 063 | 33 | 00110011 | 3 |
| 52 | 064 | 34 | 00110100 | 4 |
| 53 | 065 | 35 | 00110101 | 5 |
| 54 | 066 | 36 | 00110110 | 6 |
| 55 | 067 | 37 | 00110111 | 7 |
| 56 | 070 | 38 | 00111000 | 8 |
| 57 | 071 | 39 | 00111001 | 9 |
| 58 | 072 | 3A | 00111010 | : |
| 59 | 073 | 3B | 00111011 | ; |
| 60 | 074 | 3C | 00111100 | < |
| 61 | 075 | 3D | 00111101 | = |
| 62 | 076 | 3E | 00111110 | > |
| 63 | 077 | 3F | 00111111 | ? |
| 64 | 100 | 40 | 01000000 | @ |
| 65 | 101 | 41 | 01000001 | A |
| 66 | 102 | 42 | 01000010 | B |
| 67 | 103 | 43 | 01000011 | C |
| 68 | 104 | 44 | 01000100 | D |
| 69 | 105 | 45 | 01000101 | E |
| 70 | 106 | 46 | 01000110 | F |
| 71 | 107 | 47 | 01000111 | G |
| 72 | 110 | 48 | 01001000 | H |
| 73 | 111 | 49 | 01001001 | I |
| 74 | 112 | 4A | 01001010 | J |
| 75 | 113 | 4B | 01001011 | K |
| 76 | 114 | 4C | 01001100 | L |

| | | | | |
|---|---|---|---|---|
| 77 | 115 | 4D | 01001101 | M |
| 78 | 116 | 4E | 01001110 | N |
| 79 | 117 | 4F | 01001111 | O |
| 80 | 120 | 50 | 01010000 | P |
| 81 | 121 | 51 | 01010001 | Q |
| 82 | 122 | 52 | 01010010 | R |
| 83 | 123 | 53 | 01010011 | S |
| 84 | 124 | 54 | 01010100 | T |
| 85 | 125 | 55 | 01010101 | U |
| 86 | 126 | 56 | 01010110 | V |
| 87 | 127 | 57 | 01010111 | W |
| 88 | 130 | 58 | 01011000 | X |
| 89 | 131 | 59 | 01011001 | Y |
| 90 | 132 | 5A | 01011010 | Z |
| 91 | 133 | 5B | 01011011 | [ |
| 92 | 134 | 5C | 01011100 | \ |
| 93 | 135 | 5D | 01011101 | ] |
| 94 | 136 | 5E | 01011110 | ^ |
| 95 | 137 | 5F | 01011111 | _ |
| 96 | 140 | 60 | 01100000 | ` |
| 97 | 141 | 61 | 01100001 | a |
| 98 | 142 | 62 | 01100010 | b |
| 99 | 143 | 63 | 01100011 | c |
| 100 | 144 | 64 | 01100100 | d |
| 101 | 145 | 65 | 01100101 | e |
| 102 | 146 | 66 | 01100110 | f |
| 103 | 147 | 67 | 01100111 | g |
| 104 | 150 | 68 | 01101000 | h |
| 105 | 151 | 69 | 01101001 | i |
| 106 | 152 | 6A | 01101010 | j |
| 107 | 153 | 6B | 01101011 | k |

| | | | | |
|---|---|---|---|---|
| 108 | 154 | 6C | 01101100 | l |
| 109 | 155 | 6D | 01101101 | m |
| 110 | 156 | 6E | 01101110 | n |
| 111 | 157 | 6F | 01101111 | o |
| 112 | 160 | 70 | 01110000 | p |
| 113 | 161 | 71 | 01110001 | q |
| 114 | 162 | 72 | 01110010 | r |
| 115 | 163 | 73 | 01110011 | s |
| 116 | 164 | 74 | 01110100 | t |
| 117 | 165 | 75 | 01110101 | u |
| 118 | 166 | 76 | 01110110 | v |
| 119 | 167 | 77 | 01110111 | w |
| 120 | 170 | 78 | 01111000 | x |
| 121 | 171 | 79 | 01111001 | y |
| 122 | 172 | 7A | 01111010 | z |
| 123 | 173 | 7B | 01111011 | { |
| 124 | 174 | 7C | 01111100 | \| |
| 125 | 175 | 7D | 01111101 | } |
| 126 | 176 | 7E | 01111110 | ~ |
| 127 | 177 | 7F | 01111111 | |
| 128 | 200 | 80 | 10000000 | € |
| 129 | 201 | 81 | 10000001 | |
| 130 | 202 | 82 | 10000010 | , |
| 131 | 203 | 83 | 10000011 | *f* |
| 132 | 204 | 84 | 10000100 | „ |
| 133 | 205 | 85 | 10000101 | … |
| 134 | 206 | 86 | 10000110 | † |
| 135 | 207 | 87 | 10000111 | ‡ |
| 136 | 210 | 88 | 10001000 | ˆ |
| 137 | 211 | 89 | 10001001 | ‰ |
| 138 | 212 | 8A | 10001010 | Š |

| | | | | |
|---|---|---|---|---|
| 139 | 213 | 8B | 10001011 | ‹ |
| 140 | 214 | 8C | 10001100 | Œ |
| 141 | 215 | 8D | 10001101 | |
| 142 | 216 | 8E | 10001110 | Ž |
| 143 | 217 | 8F | 10001111 | |
| 144 | 220 | 90 | 10010000 | |
| 145 | 221 | 91 | 10010001 | ‘ |
| 146 | 222 | 92 | 10010010 | ’ |
| 147 | 223 | 93 | 10010011 | “ |
| 148 | 224 | 94 | 10010100 | ” |
| 149 | 225 | 95 | 10010101 | • |
| 150 | 226 | 96 | 10010110 | – |
| 151 | 227 | 97 | 10010111 | — |
| 152 | 230 | 98 | 10011000 | ˜ |
| 153 | 231 | 99 | 10011001 | ™ |
| 154 | 232 | 9A | 10011010 | š |
| 155 | 233 | 9B | 10011011 | › |
| 156 | 234 | 9C | 10011100 | œ |
| 157 | 235 | 9D | 10011101 | |
| 158 | 236 | 9E | 10011110 | ž |
| 159 | 237 | 9F | 10011111 | Ÿ |
| 160 | 240 | A0 | 10100000 | |
| 161 | 241 | A1 | 10100001 | ¡ |
| 162 | 242 | A2 | 10100010 | ¢ |
| 163 | 243 | A3 | 10100011 | £ |
| 164 | 244 | A4 | 10100100 | ¤ |
| 165 | 245 | A5 | 10100101 | ¥ |
| 166 | 246 | A6 | 10100110 | ¦ |
| 167 | 247 | A7 | 10100111 | § |
| 168 | 250 | A8 | 10101000 | ¨ |
| 169 | 251 | A9 | 10101001 | © |

| | | | | |
|---|---|---|---|---|
| 170 | 252 | AA | 10101010 | ª |
| 171 | 253 | AB | 10101011 | « |
| 172 | 254 | AC | 10101100 | ¬ |
| 173 | 255 | AD | 10101101 | |
| 174 | 256 | AE | 10101110 | ® |
| 175 | 257 | AF | 10101111 | ¯ |
| 176 | 260 | B0 | 10110000 | ° |
| 177 | 261 | B1 | 10110001 | ± |
| 178 | 262 | B2 | 10110010 | ² |
| 179 | 263 | B3 | 10110011 | ³ |
| 180 | 264 | B4 | 10110100 | ´ |
| 181 | 265 | B5 | 10110101 | µ |
| 182 | 266 | B6 | 10110110 | ¶ |
| 183 | 267 | B7 | 10110111 | · |
| 184 | 270 | B8 | 10111000 | ¸ |
| 185 | 271 | B9 | 10111001 | ¹ |
| 186 | 272 | BA | 10111010 | º |
| 187 | 273 | BB | 10111011 | » |
| 188 | 274 | BC | 10111100 | ¼ |
| 189 | 275 | BD | 10111101 | ½ |
| 190 | 276 | BE | 10111110 | ¾ |
| 191 | 277 | BF | 10111111 | ¿ |
| 192 | 300 | C0 | 11000000 | À |
| 193 | 301 | C1 | 11000001 | Á |
| 194 | 302 | C2 | 11000010 | Â |
| 195 | 303 | C3 | 11000011 | Ã |
| 196 | 304 | C4 | 11000100 | Ä |
| 197 | 305 | C5 | 11000101 | Å |
| 198 | 306 | C6 | 11000110 | Æ |
| 199 | 307 | C7 | 11000111 | Ç |
| 200 | 310 | C8 | 11001000 | È |

| | | | | |
|---|---|---|---|---|
| 201 | 311 | C9 | 11001001 | É |
| 202 | 312 | CA | 11001010 | Ê |
| 203 | 313 | CB | 11001011 | Ë |
| 204 | 314 | CC | 11001100 | Ì |
| 205 | 315 | CD | 11001101 | Í |
| 206 | 316 | CE | 11001110 | Î |
| 207 | 317 | CF | 11001111 | Ï |
| 208 | 320 | D0 | 11010000 | Đ |
| 209 | 321 | D1 | 11010001 | Ñ |
| 210 | 322 | D2 | 11010010 | Ò |
| 211 | 323 | D3 | 11010011 | Ó |
| 212 | 324 | D4 | 11010100 | Ô |
| 213 | 325 | D5 | 11010101 | Õ |
| 214 | 326 | D6 | 11010110 | Ö |
| 215 | 327 | D7 | 11010111 | × |
| 216 | 330 | D8 | 11011000 | Ø |
| 217 | 331 | D9 | 11011001 | Ù |
| 218 | 332 | DA | 11011010 | Ú |
| 219 | 333 | DB | 11011011 | Û |
| 220 | 334 | DC | 11011100 | Ü |
| 221 | 335 | DD | 11011101 | Ý |
| 222 | 336 | DE | 11011110 | Þ |
| 223 | 337 | DF | 11011111 | ß |
| 224 | 340 | E0 | 11100000 | à |
| 225 | 341 | E1 | 11100001 | á |
| 226 | 342 | E2 | 11100010 | â |
| 227 | 343 | E3 | 11100011 | ã |
| 228 | 344 | E4 | 11100100 | ä |
| 229 | 345 | E5 | 11100101 | å |
| 230 | 346 | E6 | 11100110 | æ |
| 231 | 347 | E7 | 11100111 | ç |

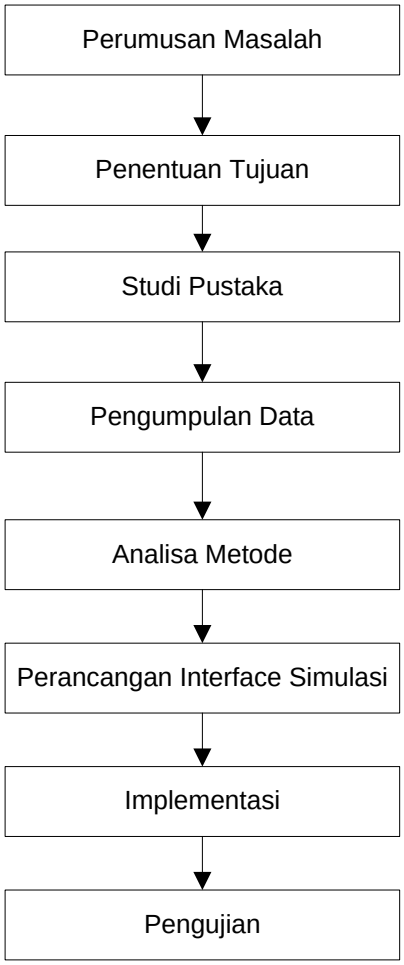
| | | | | |
|---|---|---|---|---|
| 232 | 350 | E8 | 11101000 | è |
| 233 | 351 | E9 | 11101001 | é |
| 234 | 352 | EA | 11101010 | ê |
| 235 | 353 | EB | 11101011 | ë |
| 236 | 354 | EC | 11101100 | ì |
| 237 | 355 | ED | 11101101 | í |
| 238 | 356 | EE | 11101110 | î |
| 239 | 357 | EF | 11101111 | ï |
| 240 | 360 | F0 | 11110000 | ð |
| 241 | 361 | F1 | 11110001 | ñ |
| 242 | 362 | F2 | 11110010 | ò |
| 243 | 363 | F3 | 11110011 | ó |
| 244 | 364 | F4 | 11110100 | ô |
| 245 | 365 | F5 | 11110101 | õ |
| 246 | 366 | F6 | 11110110 | ö |
| 247 | 367 | F7 | 11110111 | ÷ |
| 248 | 370 | F8 | 11111000 | ø |
| 249 | 371 | F9 | 11111001 | ù |
| 250 | 372 | FA | 11111010 | ú |
| 251 | 373 | FB | 11111011 | û |
| 252 | 374 | FC | 11111100 | ü |
| 253 | 375 | FD | 11111101 | ý |
| 254 | 376 | FE | 11111110 | þ |
| 255 | 377 | FF | 11111111 | ÿ |

# BAB III

# METODE PENELITIAN

## 3.1. Tahapan Penelitian

Adapun tahapan penelitian yang dilakukan oleh penulis ini dengan judul Perancangan Sistem Penerapan Kriptografi Keamanan Data Siswa Di SMK TR Panca Budi 1 Medan Menggunakan Metode Vigenere Chiper adalah sebagai berikut:




**Gambar 3.1. Tahapan Penelitian**

## 3.2. Metode Pengumpulan Data

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

1. Pengamatan (*Observation*)

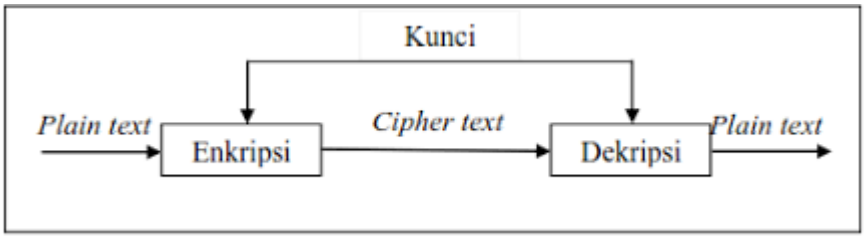
   Penulis melakukan pengamatan langsung pada setiap penggunan aplikasi chating yang sudah ada seperti WA, BBM dan Line untuk mengamati proses keamanan yang sudah dibuat sebelumnya.

2. Penelitian Kepustakaan (*Library Research*)

   Merupakan cara untuk mencari referensi dengan mengumpulkan bahan-bahan pustaka yang dilakukan di perpustakaan kampus, maupun perpustakaan umum, juga melakukan pencarian lewat internet, dengan mengunjungi situs-situs seperti *google Book online* yang dapat membantu pembahasan materi.

## 3.3. Analisis Sistem Yang Sedang Berjalan

Pertukaran data dalam hal ini pesan rahasia berbentuk teks dengan menggunakan metode tradisional yaitu dengan cara bertukar kata kunci tunggal. Diagram dibawah adalah penggambaran bagaimana pertukaran pesan rahasia menggunakan kunci tunggal terjadi.




**Gambar 3.2. Skema Pengiriman Pesan**

Pemberitahuan kata kunci dari pengirim ke penerima menggunakan media yang umum digunakan oleh banyak orang.

## 3.4. Rancangan Penelitian

Visual basic 2010 akan menjadi sarana untuk menciptakan perangkat lunak ini. Pada analisa proses ini penggunaan digunakan sebagai metode yang didalamnya terdapat kombinasi dari algoritma *Vigenere Cipher*. Algoritma *Vigenere Cipher* digunakan oleh pengirim untuk mengenkripsi pesan yang akan dikirimkan..

Perhitungan secara matematis dilakukan sebagai penggambaran proses yang akan terjadi pada metode ini yang didalamnya terdapat algoritma *Vigenere Cipher*. Berikut tahapannya:

1. Proses Enkripsi Pesan Asli oleh Pengirim

   Tahap ini dilakukan dengan menggunakan Algoritma Vigenere Cipher yang akan digunakan untuk meng-enkripsi pesan asli (*plaintext)* pengirim

Diketahui:

**Pesan = REGITA**

**Kunci = 123**

Penyelesaian:

| | |
|---|---|
| Pesan: | R |
| Kunci: | 1 |
| Chiper: | S |

| | |
|---|---|
| Pesan: | E |
| Kunci: | 2 |
| Chiper: | G |

| | |
|---|---|
| Pesan: | G |
| Kunci: | 3 |
| Chiper: | J |

| | |
|---|---|
| Pesan: | I |
| Kunci: | 1 |
| Chiper: | J |

| | |
|---|---|
| Pesan: | T |
| Kunci: | 2 |
| Chiper: | V |

| | |
|---|---|
| Pesan: | A |
| Kunci: | 3 |
| Chiper: | D |

Perhitungan Manual:

Selanjutnya akan di enkripsi dengan formula Algoritma Vigenere Cipher yaitu:

C = P + K mod 255  1

Dalam hal ini plaintext adalah ciphertext 1 yang didapat.

C1 = R + 1 mod 255

= 18 + 1 mod 255

= 19 = S

C2 = E + 2 mod 255

= 5 + 2 mod 255

= 7 = G

C3 = G + 3 mod 255

= 7 + 3 mod 255

= 10 = J

C4 = I + 1 mod 255

= 9 + 1 mod 255

= 10 = J

C5 = T + 2 mod 255

= 20 + 2 mod 255

= 22 = V

C6 = A + 3 mod 255

= 1 + 3 mod 255

= 1 = D

Maka Hasil dari enkripsi adalah:

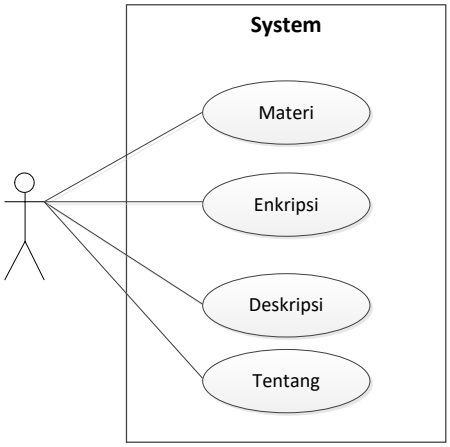
REGITA = SGJJVD

### 3.5. Perancangan Sistem

Perancangan atau Pemodelan Berorientasi Ojek merupakan proses mendapatkan informasi dari model dan menampilkannya secara grafik dengan menggunakan sebuah standar elemen grafik. Tujuan dari perancangan berorientasi objek ini memungkinkan adanya komunikasi yang lebih berkualitas antara pengguna, pengembang penganalisis, tetster, manajer dan siapapun yang terlibat dalam proyek pengembangan sistem informasi.

### 3.5.1 Use case Diagram

Berikut adalah use case diagram yang menggambarkan kegiatan.



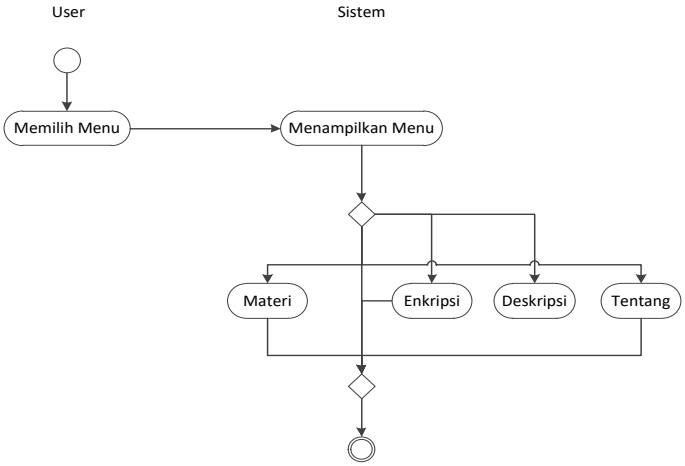

**Gambar 3.3. Use Case Diagram**

Keterangan :

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

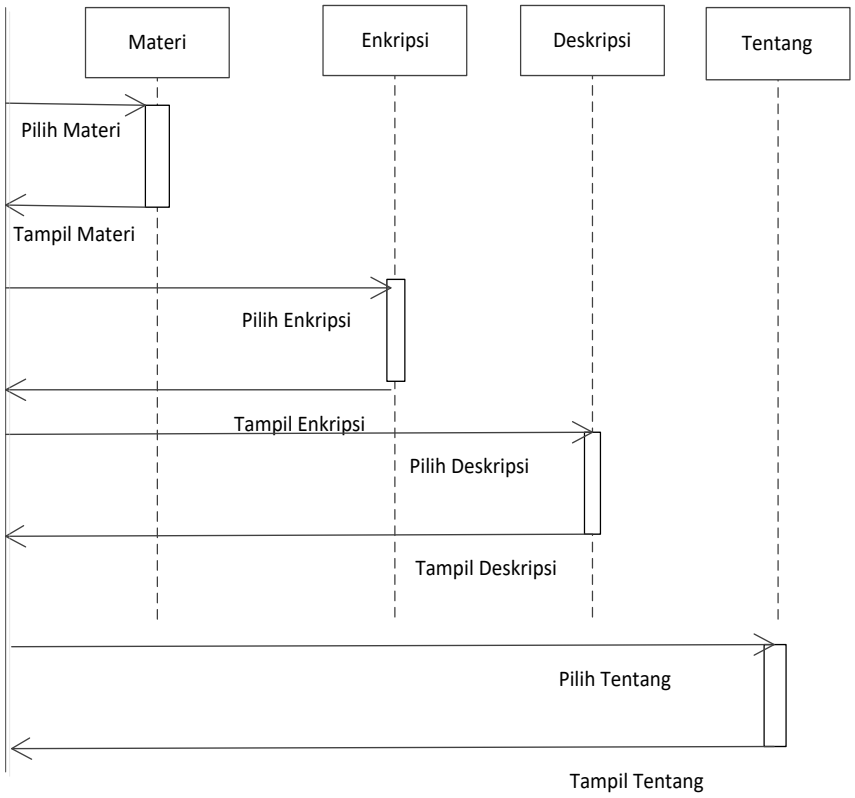
### 3.5.2 Activity Diagram

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




**Gambar 3.4. Activity Diagram**

### 3.5.3 *Sequence Diagram*




**Gambar 3.5. *Sequence Diagram***

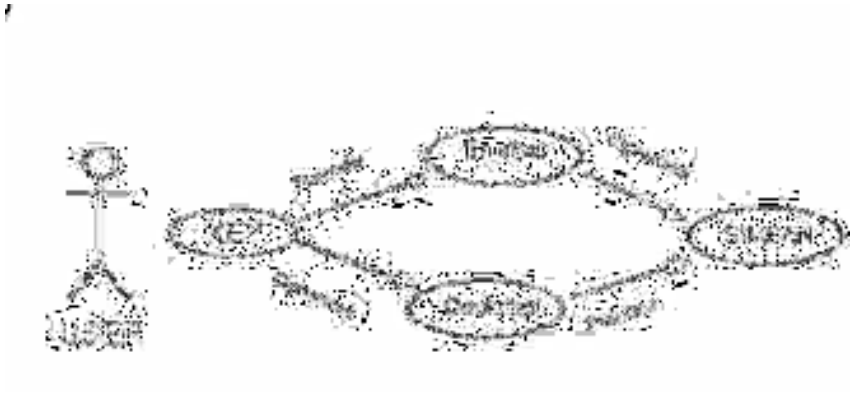
Keterangan Gambar :

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.

Use Case merupakan langkah awal pembuatan program. Dengan adanya Use Case urutan proses kegiatan menjadi lebih jelas. Bila terdapat penambahan proses maka dapat dilakukan lebih mudah. Setelah Use Case selesai disusun,selanjutnya pemrogram (programmer) menerjemahkannya ke bentuk program dengan bahasa pemrograman.

Use Case merupakan urutan-urutan langkah kerja suatu proses yang digambarkan dengan menggunakan simbo -simbol yang disusun secara sistematis.(Iswandy, 2015)

Use Case Vigenere Cipher yang digunakan oleh pengirim untuk mengenkripsi dan mendeskripsi plaintext hingga mendapatkan ciphertext digambarkan sebagai berikut:



**Gambar 3.6. Use Case Vigenere Cipher**

## 3.6. Perancangan Antarmuka

1. Rancangan Halaman Judul

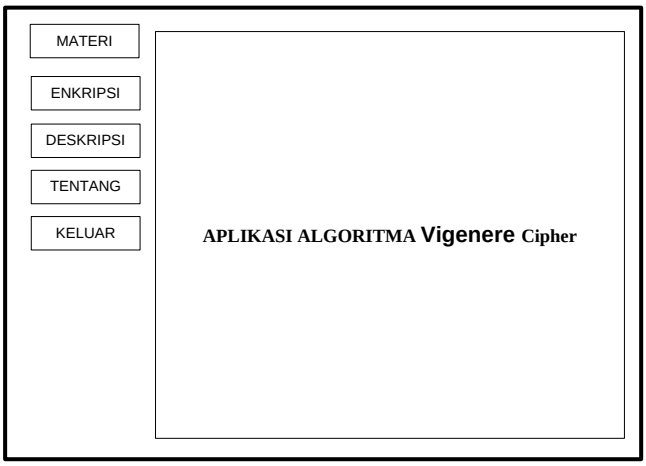
   Halaman judul merupakan halaman yang pertama muncul pada saat program dijalankan


**Gambar 3.7. Rancangan Halaman Judul**

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

2. Rancangan Halaman Menu Utama

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




**Gambar 3.8. Rancangan Halaman Menu Utama**

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

a. Tombol Materi berfungsi untuk menghubungkan pengguna ke form materi.

b. Tombol Enkripsi berfungsi untuk menghubungkan pengguna ke form Enkripsi.

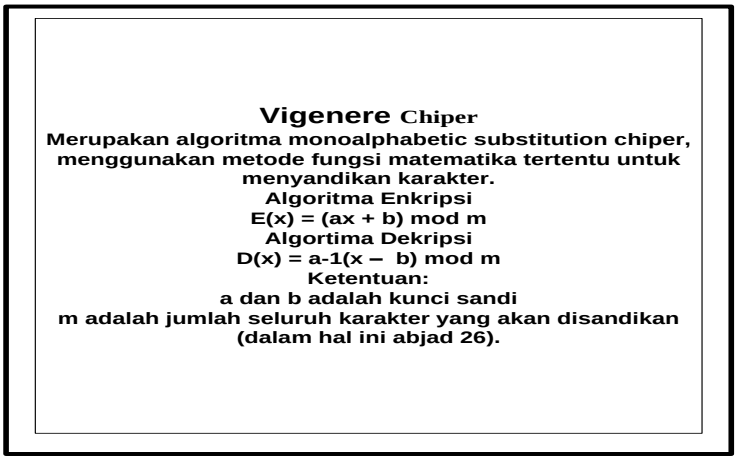
c. Tombol Deskripsi berfungsi untuk menampilkan form Deskripsi.

d. Tombol Tentang berfungsi untuk menghubungkan pengguna ke form tentang.

e. Tombol Keluar berfungsi untuk keluar dari program.

3. Rancangan Halaman Materi

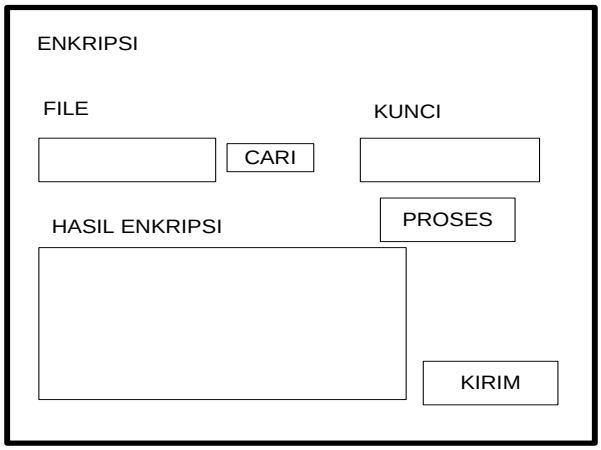
Form ini digunakan untuk menjelaskan cara kerja penyandian, dimulai dari plaintext kemudian kunci yang dikonversikan dalam bentuk angka. Setelah itu dilakukan proses penjumlahan dan jika hasil penjumlahan maka akan dikurangi 6 lalu hasilnya akan dikembalikan lagi ke dalam bentuk huruf.




**Gambar 3.9. Rancangan Halaman Materi**

4. Rancangan Halaman Enkripsi

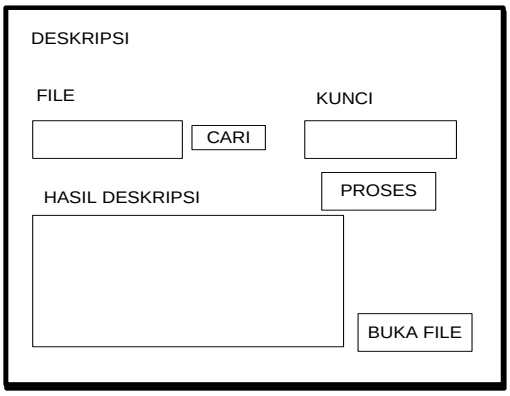
Berisi penjelasan mengenai Enkripsi. Pengguna memasukkan tulisan asli atau *plaintext* ke dalam tombol masukan *plaintext* kemudian dimasukkan juga kunci. Setelah itu, ditekan tombol Proses Enkripsi yang kemudian akan menampilkan ciphertext atau tulisan yang telah disandikan.




**Gambar 3.10. Rancangan Halaman Enkripsi**

5. Rancangan Halaman Deskripsi

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

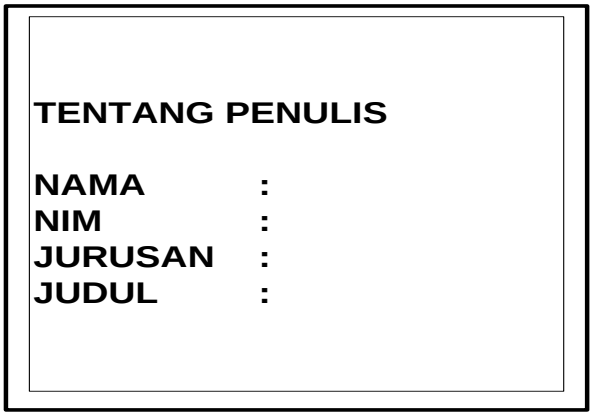



**Gambar 3.11. Rancangan Halaman Deskripsi**

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

6. Rancangan Halaman Tentang

   Berisi penjelasan mengenai tentang biodata penulis. Isi dari form tentang ini adalah berisikan data dari penulis yang ada menganggkat judul ini.




**Gambar 3.12. Rancangan Halaman Tentang**

# BAB IV

# HASIL DAN PEMBAHASAN

## 4.1. Pengujian Sistem

Pengujian system 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 berformat .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 three-pass protocol antara pengirim kepada penerima dengan kunci yang dimiliki masing-masing pihak tanpa perlu bertukar kunci tunggal hingga pada akhirnya pesan asli yang dikirimkan oleh pengirim dapat dibaca oleh penerima .

Proses yang akan dilakukan pengujian dalam aplikasi ini adalah simulasi pengiriman pesan dengan menggunakan metode algoritma *Vigenere Cipher* antara pengirim kepada penerima dengan kunci yang dimiliki masing-masing pihak tanpa perlu bertukar kunci tunggal hingga pada akhirnya pesan asli yang dikirimkan oleh pengirim dapat dibaca oleh penerima.

Tahap implementasi system merupakan tahap dimana aplikasi yang telah dirancang dijalankan. Tahap ini menunjukkan apakah setiap proses dapat berjalan

dengan baik dan mampu memberikan hasil yang diharapkan. Proses perancangan aplikasi menggunakan visual basic NET 2010 ditampilkan dalam bentuk form-form yang menjadi sarana bagi pengguna untuk melakukan proses implementasi.

### 4.1.1 Tampilan Awal / Home

Tampilan pada gambar 4.1 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.

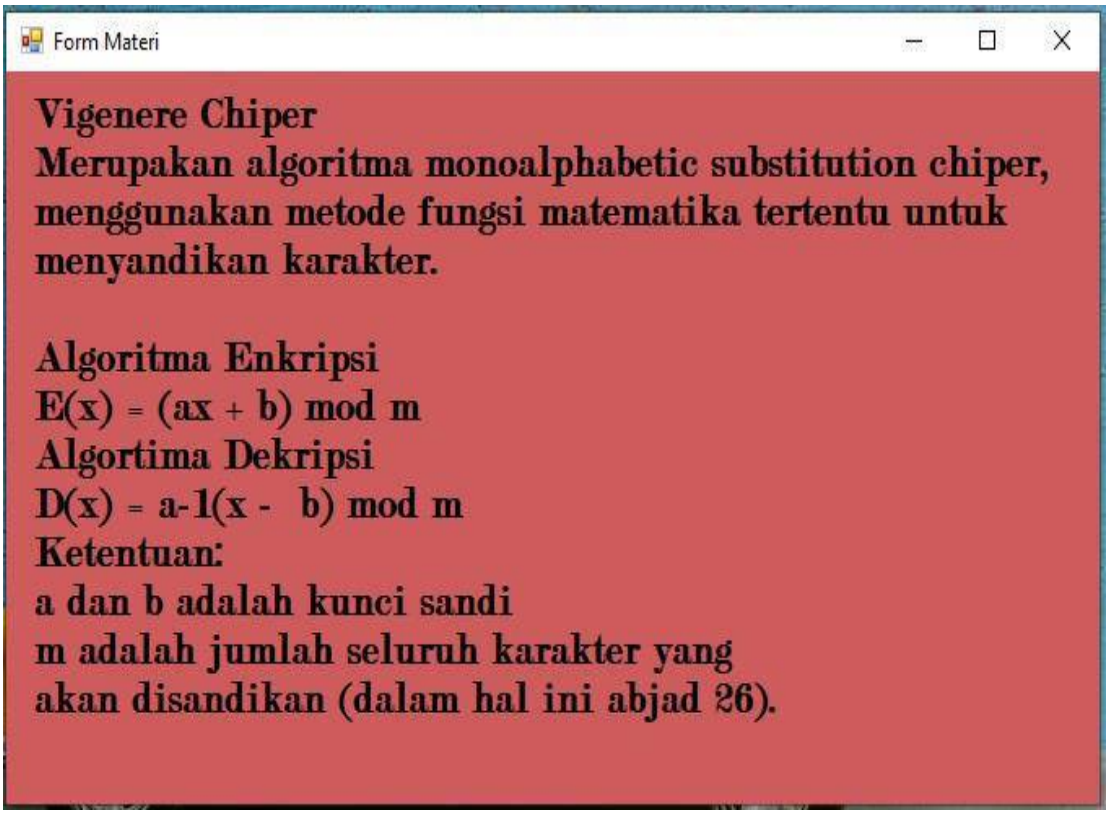

**Gambar 4.1 Tampilan Awal / Home**

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.
5. User merequest Keluar maka sistem akanKeluar.

### 4.1.2 Tampilan Aturan Materi

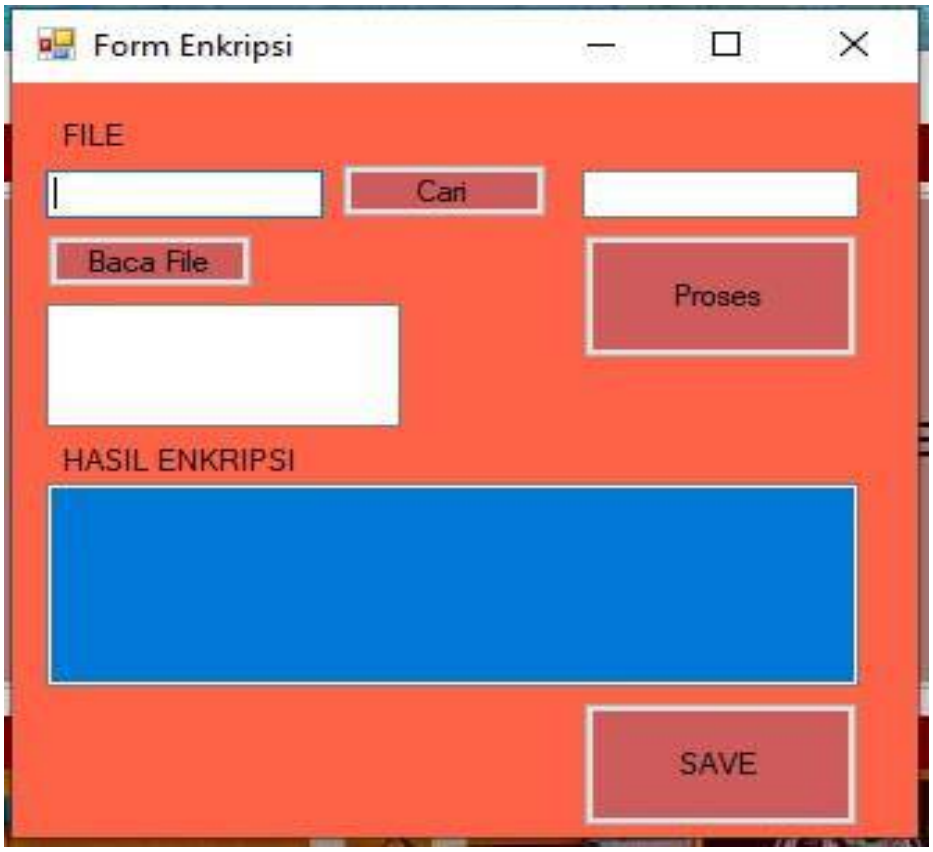
Tampilan aturan penggunaan aplikasi merupakan tampilan halaman atau form yang berisi tentang tata cara penggunaan aplikasi yang dijalankan. Padahal aman tersebut dijelaskan apa-apa saja yang menjadi kewajiban bagi pengirim dan penerima saat menjalankan simulas ialgoritma *Vigenere Cipher*.




**Gambar 4.2 Tampilan Form Materi**

### 4.1.3. Tampilan Halaman Enkripsi

Tampilan berikut merupakan tampilan pengiriman pesan pada aplikasi ini. Algoritma *Vigenere Cipher* 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 mengenkripsi pesan dan kemudian untuk mendekripsi pesan tanpa perlu mengetahui kunci yang lainnya.




**Gambar 4.3 Tampilan Halaman Enkripsi**

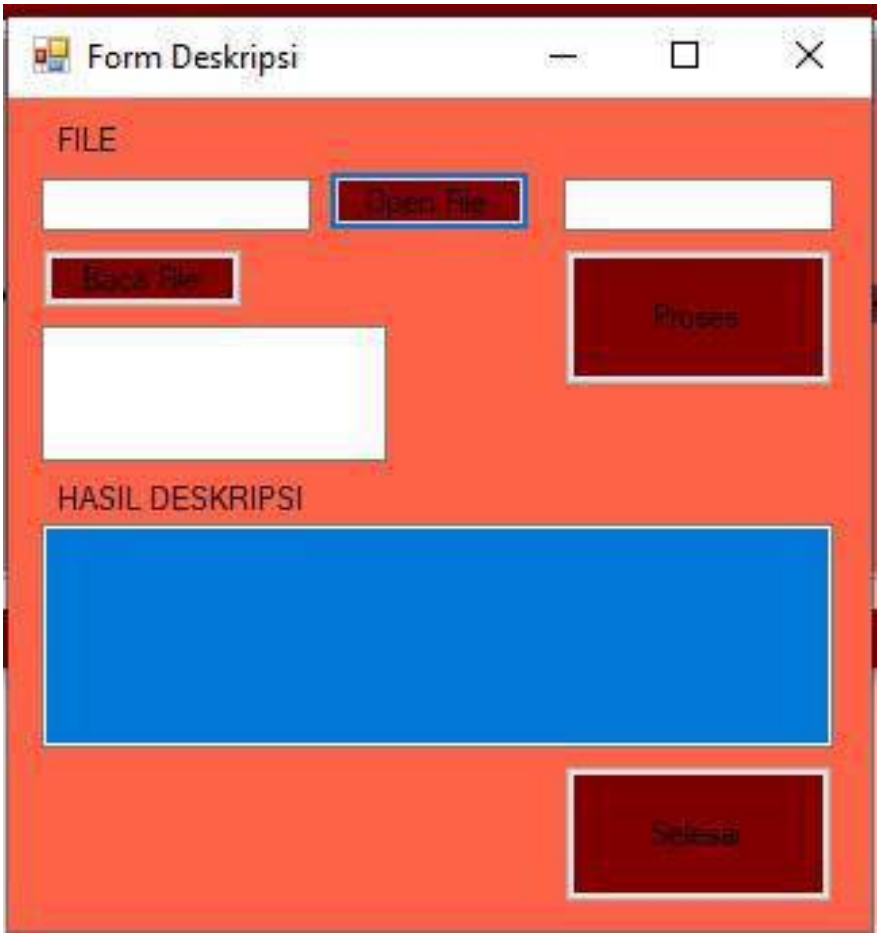
Keterangan Gambar :

1. Dalam diagram diatas menjelaskan bahwa user memilih tampilan Enkripsi maka muncul tampilan Halaman Enkripsi.
2. Dalam tampilan halaman Enkripsi user memilih File, didalam file tersebut terdapat tulisan asli atau *plaintext* kemudian user memasukkan file data Siswa yang akan di Enkripsi.
3. Kemudian user memasukkan kata sandi atau Kunci untuk memproses data tersebut.

4. Setelah itu, tekan tombol Proses maka file atau data tersebut dapat di baca di Baca File.
5. Kemudian akan muncul hasil dari Enkripsi File Data Siswa yang telah disandikan.

### 4.1.4. Tampilan Halaman Deskripsi

Tampilan berikut merupakan tampilan penerima pesan pada aplikasi ini.




**Gambar 4.4 Tampilan Halaman Deskripsi**

Keterangan Gambar :

1. Dalam diagram di atas menjelaskan bahwa user memilih Deskripsi maka mncul tampilan halaman Deskripsi.
2. Dalam tampilan halaman Deskripsi user memilih File, didalam file tersebut terdapat tulisan asli atau *plaintext* yang telah disandikan menjadi *chipertext*, kemudian user memasukkan file data Siswa yang akan di Deskripsi.
3. Kemudian user memasukkan kata sandi atau Kunci untuk memproses data tersebut.
4. Setelah itu, tekan tombol Proses maka file atau data siswa tersebut di baca di Baca File.
5. Kemudian akan muncul hasil Deskripsi untuk mengembalikan ketulisan asli jika kunci yang dimasukkan sama dengan kunci pada saat penggunaan *plaintext*.

### 4.1.5 Tampilan Halaman Tentang

Tampilan berikut adalah tentang penulis dalam halaman tersebut :

**Gambar 4.5 Tampilan Halaman Tentang**

Dalam halaman ini terdapat tentang nama penulis beserta npmnya, jurusanya dan tentang judul yang diambil penulis.

## 4.2. Pengujian *Black Box*

Perangkat lunak adalah elemen kritis dari jaminan kualitas perangkat lunak dan merepresentasikan kajian pokok dari spesifikasi, perancangan, dan pengkodean. Pengujian yang digunakan untuk menguji sistem ini adalah metode pengujian *black-box*. Pengujian *black-box* berfokus pada persyaratan fungsional perangkat lunak.

### 4.2.1. Rencana Pengujian

Pengujian fungsi Penerapan Matrix Persegi Pancajang Dalam Pengembangan Algoritma Hill Chiper dilakukan dengan menggunakan metode *Black Box*. Pengujian dilakukan pada fungsi-fungsi sistem untuk menentukan apakah fungsi tersebut telah berjalan sesuai dengan yang diharapkan.

1) Bangkitkan Kunci

**Tabel 4.1 Rencana Pengujian Tombol Cari**

| **Menu yang diuji** | **Detail pengujian** | **Kesimpulan** |
|---|---|---|
| Bankitkan Kunci | Melakukan random kunci pada proses hill chiper. | *Diterima* |

2) Proses Enkripsi

**Tabel 4.2 Rencana Pengujian Pengguna (*User*)**

| **Menu yang diuji** | **Detail pengujian** | **Jenisuji** |
|---|---|---|
| Proses | Melakukan proses enkripsi | *Diterima* |
| Kirim | Proses pengiriman file enkripsi | *Diterima* |
| Clear All | Menghapus seluruh text yang ada pada text box | *Diterima* |

3) Proses Deskripsi

**Tabel 4.3 Rencana Pengujian Pengguna (*User*)**

| **Menu yang diuji** | **Detail pengujian** | **Jenis uji** |
|---|---|---|
| Dekripsi | Melakukan proses deskripsi atau pengembalian pesan asli | *Diterima* |
| Close | Menutup semua program | *Diterima* |
| Clear All | Menghapus seluruh text yang ada pada text box | *Diterima* |

### 4.2.2. Pengujian Proses

Pengujian proses yang telah disusun, maka dapat dilakukan pengujian sebagai berikut :

**Tabel 4.4 Proses Pengujian Enkripsi dan Deskripsi (*User*)**

| **Data Pengujian Proses** | | | | | **Hasil** |
|---|---|---|---|---|---|
| **Nomor** | **Isi Pesan** | **Kunci** | **Enkripsi** | **Deskripsi** | |
| 1 | *VIGENER E*CIPHER | 5 8 | BIGLOHDKJSFU | *VIGENERE* CIPHER | Berhasil |

### 4.2.3. Kesimpulan Dan Hasil Pengujian Sistem

Hasil pengujian dari pengujian sistem telah selesai, menunjukkan bahwa sistem sudah memenuhi syarat fungsional. Secara fungsional sistem yang sudah dibangun sudah dapat menghasilkan keluaran sesuai yang diharapkan.

**Tabel 4.5 Kesimpulan Pengujian Alpha**

| **Namafungsi** | **Hasil** |
|---|---|
| Tombol Cari | Fungsi berjalandengan baik |
| Proses | Fungsi berjalan dengan baik |
| Enrkipsi | Fungsi berjalan dengan baik |
| Deskripsi | Fungsi berjalan dengan baik |
| Clear All | Fungsi berjalan dengan baik |

### 4.2.4. Kelebihan dan Kekurangan Sistem

Adapun kelebihan dan kekurangan dari system ini adalah sebagai berikut:

a. Kelebihan Sistem

- Memberikan keamanan yang lebih baik.
- Proses penginputan mudah dan friendly.
- Proses keamanan menggunakan *Viginer Cipher* yang mempersulit untuk di retas dan dirusak.

b. Kekurangan Sistem

- Hanya melakukan enkripsi dan deskripsi.
- Tidak bisa menggunakan terlalu banyak karakter.

# BAB V

# PENUTUP

## 5.1. Kesimpulan

Berdasarkan pembahasan dalam Perancangan sistem Penerapan Kriptografi Keamanan Data Siswa di SMK TR Panca Budi 1 Medan Menggunakan Metode *Vigenere Chiper*, maka dapat diambil kesimpulan sebagai berikut :

1. Dengan adanya sistem keamaman data pada file Ms. Word di SMK TR Panca Budi 1 Medan, data siswa menjadi sulit dibuka oleh pihak ketiga (orang yang tidak bertanggung jawab) dikarenakan file sudah dilakukan enkripsi dengan menggunakan kriptografi *vigenere chiper*.
2. Aplikasi yang dibuat dalam proses keamanan data siswa juga dapat membantu proses penginputan data siswa.

## 5.2. 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


Anonim, E. H. Rachmawanto and C. A. Sari, "Keamanan File Menggunakan Teknik Kriptografi Shift Cipher," Jurnal Techno. Com, vol. 14, no. 2, pp. 329-335, 2014.

Bishop, Rosdiana, "*Sekuritas Sistem Dengan Kriptografi*," in Prosiding Sendi_U 2013, Semarang, 2013.

Fachri, barany, agus perdana windarto, and ikhsan parinduri. "penerapan backpropagation dan analisis sensitivitas pada prediksi indikator terpenting perusahaan listrik." jepin (jurnal edukasi dan penelitian informatika) 5.2 (2019): 202-208.

Fachri, b., windarto, a. P., & parinduri, i. (2019). Penerapan backpropagation dan analisis sensitivitas pada prediksi indikator terpenting perusahaan listrik. Jepin (jurnal edukasi dan penelitian informatika), 5(2), 202-208.

Fachri, barany; windarto, agus perdana; parinduri, ikhsan. Penerapan backpropagation dan analisis sensitivitas pada prediksi indikator terpenting perusahaan listrik. Jepin (jurnal edukasi dan penelitian informatika), 2019, 5.2: 202-208.

Hamdi, nurul. "model penyiraman otomatis pada tanaman cabe rawit berbasis programmable logic control." jurnal ilmiah core it: community research information technology 7.2 (2019).

Haviluddin, H. (2015). Memahami Penggunaan UML (Unified Modelling Language). Informatika Mulawarman: Jurnal Ilmiah Ilmu Komputer, 6(1), 1-15.

Munir, F. A. (2014). Perancangan aplikasi pengamanan data dengan kriptografi Advanced Encryption Standard (AES). Pelita Informatika: Informasi dan Informatika, 4(1).

Nandar Pabokory, Indah Fitri Astuti, Awang Harsa Kridalaksana, " *Implementasi Kriptografi Pengamanan Data Pada Pesan Teks, Isi File Dokumen, Dan File Dokumen Menggunakan Algoritma Advanced Encryption Standard*," Jurnal Informatika Mulawarman Vol. 10. Nomor 1, 2015.

Notoatmodjo, H., Nuryanti, N., Fera, V. V., Warsinah, W., & Sholihat, N. K. (2018). Pengaruh Edukasi Terhadap Pengetahuan, Sikap, Dan Kemampuan Berkomunikasi Atas Informasi Obat. Kartika: Jurnal Ilmiah Farmasi, 4(1), 10-15.

Permana, aminuddin indra. "kombinasi algoritma kriptografi one time pad dengan generate random keys dan vigenere cipher dengan kunci em2b." (2019).

Putra, randi rian. "sistem informasi web pariwisata hutan mangrove di kelurahan belawan sicanang kecamatan medan belawan sebagai media promosi." jurnal ilmiah core it: community research information technology 7.2 (2019).

Putra, randi rian, et al. "decision support system in selecting additional employees using multi-factor evaluation process method." (2019).

Putra, randi rian. "implementasi metode backpropagation jaringan saraf tiruan dalam memprediksi pola pengunjung terhadap transaksi." jurti (jurnal teknologi informasi) 3.1 (2019): 16-20.

Rhee, C. A. Sari, E. H. Rachmawanto, Y. P. Astuti and L. Umaroh, "Optimasi Penyandian File Kriptografi Shift Cipher," in Prosiding Sendi_U 2013, Semarang, 2013.

Renddy, Teady Matius, Surya Mulyana, Fresly, " *Steganografi Dengan Deret Untuk Mengacak Pola Penempatan Pada Rgb*," Jurnal Teknologi Informasi, 2015.

Safik, M. (2018). Penggunaan Media Gambar Berseri dalam Meningkatkan Kemampuan Menulis Bahasa Jepang Siswa Kelas XII pada Kelas Bahasa MAN Model Manado. Jurnal Ilmiah Iqra', 7(2).

Saputra, muhammad juanda, and nurul hamdi. "rancang bangun aplikasi sejarah kebudayaan aceh berbasis android studi kasus dinas kebudayaan dan pariwisata aceh." journal of informatics and computer science 5.2 (2019): 147-157

Sholeh, M., & Hamokwarong, J. V. (2014). Aplikasi Kriptografi Dengan Metode Vernam Cipher dan Metode Permutasi Biner. JURNAL ILMIAH MOMENTUM, 7(2).

Sidik, a. P., efendi, s., & suherman, s. (2019, june). Improving one-time pad algorithm on shamir's three-pass protocol scheme by using rsa and elgamal algorithms. In journal of physics: conference series (vol. 1235, no. 1, p. 012007). Iop publishing.

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.

Tasril, v., wijaya, r. F., & widya, r. (2019). Aplikasi pintar belajar bimbingan dan konseling untuk siswa sma berbasis macromedia flash. Jurnal informasi komputer logika, 1(3).

Urva, G., & Siregar, H. F. (2015). Pemodelan UML E-Marketing Minyak Goreng. JURTEKSI ROYAL Edisi2.

Yulansari, K. (2013, March). Sistem Informasi Pengolahan Data Iuran Badan Pembantu Penyelenggaraan Pendidikan Sekolah Menengah Kejuruan Negeri 2 Donorojo. In Seruni-Seminar Riset Unggulan Nasional Inoformatika dan Komputer (Vol. 2, No. 1).