PENERAPAN MD5 PADA VERIFIKASI DAN VALIDASI KEASLIAN DATA

Disusun dan Diajukan Untuk Memenuhi Persyaratan Ujian Akhir
Memperoleh Gelar Sarjana Komputer Pada Fakultas Saiñs dan Teknologi
Universitas Pembangunan Panca Budi
Medan

SKRIPSI

OLEH :

NAMA	**: RAHADIYAN BAYU ANJASWORO**
N.P.M	**: 1414370329**
PROGRAM STUDI	**: SISTEM KOMPUTER**

FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS PEMBANGUNAN PANCA BUDI
MEDAN
2019

LEMBAR PENGESAHAN

PENERAPAN MD5 PADA VERIFIKASI DAN VALIDASI KEASLIAN DATA

Disusun Oleh :

NAMA	:	**RAHADIYAN BAYU ANJASWORO**
N.P.M	:	**1414370329**
PROGRAM STUDI	:	**SISTEM KOMPUTER**

Skripsi Telah Disetujui oleh Dosen Pembimbing Skripsi
Pada Tanggal : 08 November 2019

Dosen Pembimbing I	Dosen Pembimbing II
Hysah P. U. Siahaan S.Kom, M.Kom, Ph.D	**Rian Farta Wijaya S.Kom, M.Kom**

Mengetahui,

Dekan Fakultas Sains dan Teknologi	Ketua Program Studi Sistem Komputer
Sri Shindi Indira, S.T., M.Sc	**Eko Hariyanto, S.Kom., M.Kom**

SURAT PERNYATAAN

Saya yang bertanda tangan di bawah ini :

Nama	: Rahadiyan Bayu Anjasworo
NPM	: 1414370329
Prodi	: Sistem Komputer
Konsentrasi	: Keamanan Jaringan Komputer (KJK)
Judul Skripsi	: Penerapan MD5 pada Verifikasi danValidasi Keaslian Data

Dengan ini mengajukan permohonan untuk ujian sarjana lengkap pada Fakultas Sains & Teknologi Universitas Pembangunan Pancabudi.

Sehubungan dengan hal tersebut, maka saya tidak akan lagi ujian perbaikan nilai dimasa yang akan datang.

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

Medan, November 2019

Yang membuat pernyataan

RAHADIYAN BAYU ANJASWORO

SURAT PERNYATAAN

Saya yang bertanda tangan di bawah ini :

Nama	: Rahadiyan Bayu Anjasworo
NPM	: 1414370329
Prodi	: Sistem Komputer
Konsentrasi	: Keamanan Jaringan Komputer (KJK)
Judul Skripsi	: Penerapan MD5 pada Verifikasi danValidasi Keaslian Data

Dengan ini menyatakan bahwa :

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

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

Medan, November 2019

Yang membuat pernyataan

RAHADIYAN BAYU ANJASWORO

UNIVERSITAS PEMBANGUNAN PANCA BUDI
FAKULTAS SAINS & TEKNOLOGI

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

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

PERMOHONAN MENGAJUKAN JUDUL SKRIPSI

ng bertanda tangan di bawah ini :

engkap	: RAHADIYAN BAYU ANJASWORO
Tgl. Lahir	: MEDAN / 06 Januari 1997
Pokok Mahasiswa	: 1414370329
Studi	: Sistem Komputer
rasi	: Keamanan Jaringan Komputer
Kredit yang telah dicapai	: 141 SKS, IPK 3.23

ni mengajukan judul skripsi sesuai dengan bidang ilmu, dengan judul:

Judul SKRIPSI	Persetujuan
erancangan DNS Server menggunakan Cloud Flare	☐
erapan MD5 pada verifikasi pesan terenkripsi dengan Metode Hill Cipher	☑
plementasi menggunakan teknik enkripsi berbasis Visual Basic Net	☐

ng disetujui oleh Kepala Program Studi diberikan tanda ☑

Judul: penerapan MD5 pada verifikasi dan validasi keaslian data.

Medan, 05 September 2018

Rektor I,

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

Pemohon,

(Rahadiyan Bayu Anjasworo)

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

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

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

(.....Audyna Putri..)

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

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

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

(.....Rizki Rahma Wati..)

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

sitas : Universitas Pembangunan Panca Budi
as : SAINS & TEKNOLOGI
Pembimbing I : Andysah Putera Utama Siahaan, S. Kom., M. Kom
Pembimbing II : Ryan Parta Wijaya, S. Kom.
Mahasiswa : RAHADIYAN BAYU ANJASWORO
n/Program Studi : Sistem Komputer
Pokok Mahasiswa : 1414370329
g Pendidikan : S1
Tugas Akhir/Skripsi : Penerapan MDS pada Verifikasi pesan terenkripsi dengan Metode Hill Chiper

NGGAL	PEMBAHASAN MATERI	PARAF	KETERANGAN
1 2019	Rev Judul		
	Acc Seminar Judul		
1	Rev Bab I		
2	Rev Bab II		
5	Rev Bab III		
7	Rev Bab IV		
7	Rev Bab IV, V		
	Acc Seminar		
10	Acc Sidang		
11	Acc Jilid		

Medan, 08 Januari 2019
Diketahui/Disetujui oleh :
Dekan,

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

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

sitas : Universitas Pembangunan Panca Budi
as : SAINS & TEKNOLOGI
Pembimbing I : Andysah Putera Utama Siahaan, S.Kom., M.Kom
Pembimbing II : Ryan Parta Wijaya, S.Kom., M.Kom
Mahasiswa : RAHADIYAN BAYU ANJASWORO
n/Program Studi : Sistem Komputer
Pokok Mahasiswa : 1414370329
g Pendidikan : S1
Tugas Akhir/Skripsi : Penerapan MDS pada Verifikasi Pesan Terenkripsi dengan Metode Hill Cipher

NGGAL	PEMBAHASAN MATERI	PARAF	KETERANGAN
.1.2019	Diskusi judul & Bab 1 & ACC Seminar judul		
8.2019	Bab 2 (print jurnal & tambahkan analisa penerapan MDS di Bab 3)		
.9.2019	Bab 3		
.10.2019	Acc bab 4,5 Acc seminar hasil		
10.2019	Acc Sidang		
11.2019	Acc jilid		

Medan, 08 Januari 2019
Diketahui/Disetujui oleh :
Dekan,

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

Telah Diperiksa oleh LPMU
dengan Plagiarisme..... 59 %
Medan 23 OKTOBER 2019
AN Ka. LPMU
HUSNI M. RITONGA, BA., MSc.
Calvo Pramono, SE.,M.M

FM-BPAA-2012-0

l : Permohonan Meja Hijau

Medan, 23 Oktober 2019
Kepada Yth : Bapak/Ibu Dekan
Fakultas SAINS & TEKNOLOGI
UNPAB Medan
Di -
Tempat

Telah di terima
berkas persyaratan
proses
Medan 24 / 10 / 2019
An. KA. BPAA
TEGUH WAHYONO, SE., MM

ngan hormat, saya yang bertanda tangan di bawah ini :

ma	: RAHADIYAN BAYU ANJASWORO
mpat/Tgl. Lahir	: MEDAN / 06 januari 1997
ma Orang Tua	: R Koesno Utoyo
P. M	: 1414370329
kultas	: SAINS & TEKNOLOGI
ogram Studi	: Sistem Komputer
. HP	: 082165301461
amat	: Jl. Flores Gg. Prona Lk.IV

tang bermohon kepada Bapak/Ibu untuk dapat diterima mengikuti Ujian Meja Hijau dengan judul **Penerapan MD5 pada verifikasi dan lidasi Keaslian Data**, Selanjutnya saya menyatakan :

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

1.	[102] Ujian Meja Hijau	: Rp.	250.000
2.	[170] Administrasi Wisuda	: Rp.	1,500,000
3.	[202] Bebas Pustaka	: Rp.	100,000
4.	[221] Bebas LAB	: Rp.	5,000
	Total Biaya	: Rp.	1,805,000
	F. UK 50 %	Rp.	2,750.000

24 / OKTOBER 2019

Ukuran Toga : XXL

Rp. 4.500.000

Ketahui/Disetujui oleh
Dekan Fakultas SAINS & TEKNOLOGI
Shindi Indira, S.T., M.Sc.

Hormat saya
RAHADIYAN BAYU ANJASWORO
1414370329

tan :

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

TANDA BEBAS PUSTAKA
No. 943 / Perp / BP / 2019
Telah tidak ada sangkut paut dengan UPT. Perpustakaan
Medan 23 OCT 2019
Kepala UPT Perpustakaan
M. Muttaqin, S.kom, M.kom

UNIVERSITAS PEMBANGUNAN PANCA BUDI
UNPAB INDONESIA
UPT. PERPUSTAKAAN

UKM CENTER
PANCA BUDI

Plagiarism Detector v. 1092 - Originality Report:

Analyzed document: 09/10/2019 08:33:54

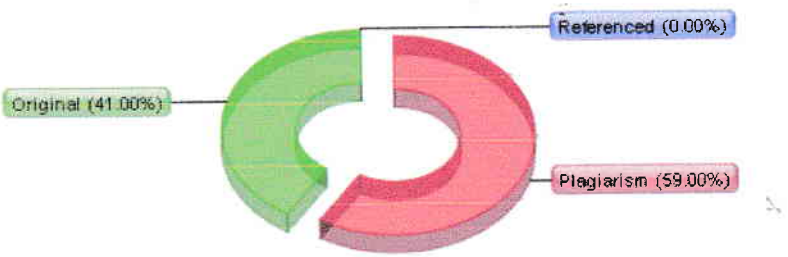
"RAHADIYAN BAYU ANJASWORO_1414370329_SISTEM KOMPUTER.docx"

Licensed to: **Universitas Pembangunan Panca Budi_License4**

UNIVERSITAS PEMBANGUNAN PANCA BUDI
Beriman, Berilmu, Berkarya

LEMBAGA PENJAMIN MUTU UNIVERSITAS PEMBANGUNAN PANCA BUDI
3/@23/10-19

Relation chart:



Distribution graph:

Comparison Preset: Rewrite. Detected language: Indonesian

Top sources of plagiarism:

% 13	wrds: 1242	https://pastebin.com/FNPwTtbP
% 12	wrds: 805	http://eprints.binadarma.ac.id/258/1/JURNAL%20STUDI%20DAN%20IMPLEMENTASI%20PENGAMANAN%20BA...
% 8	wrds: 708	https://lb.wikipedia.org/wiki/American_Standard_Code_for_Information_Interchange

ow other Sources:]

Processed resources details:

297 - Ok / 27 - Failed

ow other Sources:]

Important notes:

Wikipedia:	Google Books:	Ghostwriting services:	Anti-cheating:
WIKIPEDIA!	Google		

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

KARTU BEBAS PRAKTIKUM

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

ama	:	RAHADIYAN BAYU ANJASWORO
P.M.	:	1414370329
ngkat/Semester	:	Akhir
kultas	:	SAINS & TEKNOLOGI
rusan/Prodi	:	Sistem Komputer

nar dan telah menyelesaikan urusan administrasi di Laboratorium Komputer Universitas Pembangunan Panca Budi Medan.

Medan, 22 Oktober 2019
a.n. Ka. Laboratorium

Fachrid Wadly, S. Kom

ABSTRAK

RAHADIYAN BAYU ANJASWORO

Penerapan MD5 Pada Verifikasi dan Validasi Keaslian Data

2019

Keamanan data dan informasi merupakan hal yang sangat penting di era informasi saat ini. Umumnya, setiap institusi memiliki dokumen-dokumen penting dan bersifat rahasia yang hanya boleh diakses oleh orang tertentu. Sistem informasi yang dikembangkan harus menjamin keamanan dan kerahasian dokumen-dokumen tersebut. Namun kendalanya bahwa media-media yang digunakan sering kali dapat disadap oleh pihak lain. Oleh karena itu, diperlukan metode untuk mengamankannya, salah satunya dengan menggunakan metode kriptografi modern. Berdasarkan latar belakang diatas maka penulis tertarik untuk memilih judul "Penerapan MD5 Pada Verifikasi dan Validasi Keaslian Data". Kesimpulan berdasarkan pembahasan dalam perancangan Penerapan MD5 dan Validasi Keaslian Data, maka dapat diambil kesimpulan sebagai berikut Perangkat lunak ini dirancang untuk mengamankan file data text pada proses Verifikasi menggunakan metode MD5, Penggunaan metode MD5 sangat baik digunakan untuk proses pengamanan file, Penggunaan kunci sulit di tebak dikarenakan menggunakan hexdecimal to binary.

Kata Kunci: Kriptografi, MD5.

DAFTAR ISI

Halaman

DAFTAR GAMBAR

Halaman

DAFTAR TABEL

Halaman

BAB I

PENDAHULUAN

1.1 Latar Belakang

Keamanan data dan informasi merupakan hal yang sangat penting di era informasi saat ini. Umumnya disetiap institusi memiliki dokumen yang penting dan bersifat rahasia yang hanya boleh diakses oleh orang tertentu. Sistem informasi yang dikembangkan harus menjamin keamanan dan kerahasian dokmen tersebut. Namun kendalanya bahwa media-media yang digunakan sering kali dapat disadap oleh pihak lain. Oleh karena itu diperlukan metode untuk mengamankannya salah satunya dengan menggunakan metode *kriptografi modern*.

Dalam *kriptografi moder*n, penulis ini membuat keamana data menggunakan metode MD5. Proses pengamanan data tersebut hanya berupa *text*, angka dan simbol yang dikirim dan penerima harus memiliki kunci untuk membuka data asli. Dengan adanya metode MD5 data teks yang muncul barupa hasil dari metode tersebut saat ini, ilmu *kriptografi modern* semakin banyak digunakan dan mulai berubah menjadi kebutuhan. Dengan maraknya perkembangan ilmu dan teknologi. Informasi-informasi yang penting pun tidak lagi hanya berada pada media tulis saja.

Bedasarkan penelitian terdahulu yang telah dilakukan oleh (saipul bahri, 2016) dengan judul studi Implementasi Pengamanan Basis Data Menggunakan

Metode Enkripsi MD5 dapat menyimpulkan bahwa masalah keamanan merupakan salah satu tantangan yang harus dapat terjamin keamanannya. Pengamanan data dapat dilakukan melalui dua cara. Cara pertama ialah pengamanan data dari sisi kandungan data yang tersimpan pada basi data. Makalah ini menguraikan implementasi pengamanan data pada basis data dengan cara kedua. Pengamanan data dilakukan dengan menggunakan teknik kriptografi MD5. Penelitian (studi) yang dilakukan ialah untuk mencari cara agar MD5 dapat dimanfaatkan untuk mengamankan data serta memberi kemuudahan bagi pemilik data uuntuk mengamankan data tanpa perlu mengetahui *query-query* yang perlu diketikkan atau dijalankan.

Keamanan dari enkripsi konvensional bergantung pada beberapa faktor. Pertama algoritma enkripsi harus cukup kuat sehingga menjadikan sangat sulituntuk mendekripsi cipher taks dengan dasar *cipher teks* tersebut. Lebih jauh dari itu keamanan dari algoritma *enkripsi konvensional* bergantung pada kerahasian dari kuncinya bukan algoritmanya, yaitu dengan asumsi bahwa sangat tidak praktis untuk mendekripsikan informasi dengan dasar cipher teks dan pengetahuan tentang algoritma enkripsi / diskripsi. Atau dengan kata lain, kita tidak perlu menjaga kerahasiaan dari algoritma tetapi cukup dengan kerahasiaan kuncinya.

Berdasarkan pembahasan diatas, Penulis akan membuat suatu aplikasi penerapan metode MD5 dengan menggunakan sistem yang berbasiskan Desktop. Aplikasi yang akan penulis rancang adalah sebagai penerapan metode MD5 agar dapat memahami cara teknik enkripsi dan dekripsi data yang digunakan kepada

pengguna yang masih awam dalam teknik manipulasi data tersebut. Berdasarkan latar belakang diatas makan penulis tertarik untuk memilih judul "**Penerapan MD5 pada Verifikasi danValidasi Keaslian Data**"**.**

1.2 Rumusan Masalah

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

1. Bagaimana merancang sebuah *software* dalam proses Verifikasi dan Keaslian Data menggunakan metode MD5 sebagai verifikasi data?
2. Bagaimana menerapkan *kriptografi modern* pada proses verifikasi sebuah data?

1.3 Batasan Masalah

Dalam perancangan aplikasi pengamanan informasi ini penulis membatasi masalah sebagai berikut :

1. Aplikasi yang dibangun hanya melakukan proses verifikasi dan keaslian data.
2. Perancangan aplikasi merupakan simulasi secara dekstop.
3. Program yang digunakan dalam perancangan aplikasi ini adalah *Visual Studio 2010* menggunakan metode MD5 dalam proses verifikasi data.

1.4 Tujuan Penelitian

Tujuan yang ingin dicapai penulis dalam perancangan aplikasi penerapan metode *MD5* ini adalah :

1. Merancang aplikasi keamanan informasi data dengan menggunakan metode MD5.
2. Merancang sistem pengamanan informasi data dengan proses verifikasi menggunakan metode metode MD5.
3. Menciptakan sebuah software yang khusus dalam Verifikasi dan Keaslian Data.

1.5 Manfaat Penelitian

Perancangan aplikasi penerapan *metode* MD5 ini bermanfaat bagi masyarakat luas antara lain :

1. Dengan menggunakan aplikasi ini seseorang dapat mengamankan suatu informasi tanpa takut diketahuin oleh orang lain.
2. Dapat digunakan dalam proses kerahasian data.
3. Proses pertukaran data atau informasi menjadi aman.

BAB II

LANDASAN TEORI

2.1 Kemanan Data

Pada zaman teknologi informasi sekarang, data atau informasi merupakan 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 muuncul maasalah 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 metugikan ihak tertentu. Dalam keamanan data ada beberapa aspek yang berkaitan dengan persyaratan keamanan 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. (Fresly, 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). (Fresly, 2015). *File* yang dapat di*Enkripsi* dapat berupa teks, gambar maupun audio dan video.

2.3 Macam-Macam *Kriptografi*

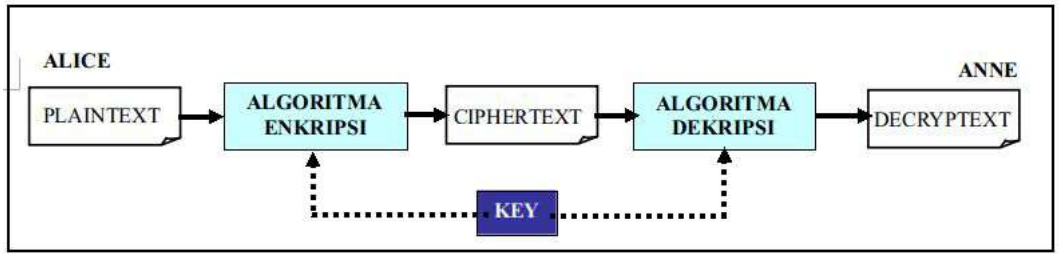
Kriptografi dibedakan menjadi 3 bagian yaitu *kriptografi* simetris, *kriptografi* asimetris dan fungsi hash satu arah. *Kriptografi* simetris disebut juga *kriptografi* kunci rahasia merupakan jenis *kriptografi* paling intuitif. Ini termasuk penggunaan kunci rahasia yang dikenal hanya pada pengguna komunikasi yang aman. *Kriptografi* asimetris sendiri berbeda dengan *kriptografi* simetris, dimana *kriptografi* asimetris ini menggunakan dua kunci yang berbeda, yaitu kunci publik dan kunci rahasia atau kunci pribadi. Kunci-kunci tersebut berhubungan secara matematis, tetapi tidak mungkin secara perhitungan untuk menarik kesimpulan satu dengan yang lain.

Fungsi *hash* satu arah, juga dikenal sebagai rangkuman pesan atau fungsi kompresi adalah fungsi matematis yang mengambil input panjang variabel dan mengubahnya ke dalam urutan biner dengan panjang yang tetap.

2.4 *Kriptografi Simetris*

Pada *Kriptografi Simetris*, kunci yang digunakan pada proses enkripsi dan dekrpsi bernilai sama. Proses yang dilakukan dalam kriptografi simetris terbagi atas dua jenis yaitu substitusi dan transposisi. Pada proses subtitusi, setiap karakter di ganti dangan karakter lain, sementara pada transposisi, setiap karakter

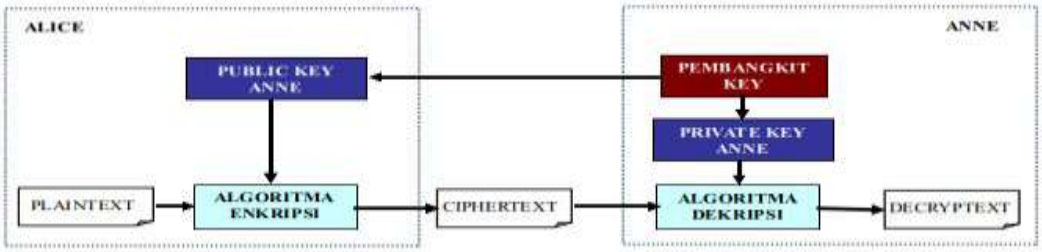
bertukar posisi dengan karakter lain. Hal yang harus diperhatikan pada penggunaan kriptografi simetris adalah keamanan kunci. Jika kunci diketahui oleh pihak lain, maka *cipherext* yang dirahasiakan dapat dipecahkan. Berikut ini adalah gambar dari proses penggunaan kunci simetris.



Gambar 2.1 Alur *Kriptografi Simetris*

2.5 *Kriptografi Asimetris*

Pada *Kriptodrafi Asimetris* kuunci yang digunakan pada proses enkripsi berbeda dengan kunci yang digunakan untuk proses dekripsi. Kunci enkripsi disebut kunci publik dan kunci ntuk dekripsi disebut dengan kunci *private*. Oleh karena itu, algoritma ini disebut juga dengan *kriptografi* kunci publik (public key). Berikut ini adalah gambar dari proses penggunaan kunci asimetris, yang ditampilkan pada gambar dibawah ini.



Gambar 2.2 Alur *Kriptografi Asimetris* (Sadikin, 2012)

Kelebihan Kriptografi Kunci-Publik (*Asimetri)*:

1. Hanya kunci *privat* yang perllu dijaga kerahasiaannya oleh setiap entitas yang berkomunikasi (tetapi, otentikasi kunci publik tetap harus terjamin). Tidak ada kebutuhan mengirim kunci privat sebagaimana pada sistem simetri.
2. Pasangan kunci publik/kunci privat perlu diubah, bahkan dalam periode waktu yang panjang.
3. Dapat digunakan untuk menggambarkan pengiriman kunci simetri.
4. Beberapa algoritma kunci-publik dapat digunakan untuk memberi tanda tangan digital pada pesan.

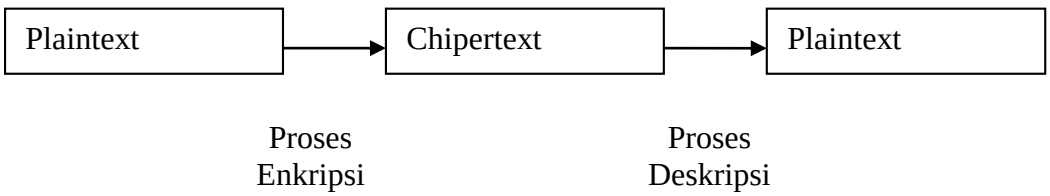
Kelemahan kriptografi kunci-publik *(asimetri*):

1. Enkripsi dan dekripsi data umumnya lebih lambat daripada sistem simetri, karena *enkripsi* dan *dekripsi* menggunakan bilangan yang besar dan melibatkan operasi perpangkatan yang besar.
2. Ukuran cipherteks lebih besar daripada *plainteks* (bisa dua sampai empat kali ukuran plainteks).
3. Ukuran kunci relatif lebih besar daripada ukuran kunci simetri.
4. Karena kunci publik diketahui secara luas dan dapat digunakan setiap orang, maka *cipherteks* tidak memberikan informasi mengenai otentikasi pengirim.
5. Tidak ada algoritma kunci-publik yang terbukti aman (sama seperti *block cipher*). Kebanyakan algoritma mendasarkan keamanannya pada sulitnya

memecahkan persoalan-persoalan *aritmetik* (logaritmik, pemfaktoran dan sebagainya) yang menjadi dasar pembangkitan kunci.

2.6 *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 2.3. Proses *Enkripsi* dan *Deskripsi*

(Sumber: Fresly, 2015)

2.7 Kriptografi Klasik

Menurut (bishop, 2014). *Kriptografi* klasik adalah *kriptografi* yang disebut juga sebagai *kriptografi* kunci tunggal atau *kriptografi simetris* menggunakan

kunci yang sama untuk *Enkripsi* maupun *Deskripsi*. *Kriptografi* klasik merupakan *kriptografi* yang digunakan pada zaman dahulu sebelum komputer ditemukan namun belum secanggih sekarang. *Kriptografi* ini melakukan pengacakan huruf pada kata terang / *plaintext*.

2.8 *Algoritma Merkle Hellman*

Merupakan salah satu *algoritma kriptografi* kunci-public awal yang ditemukan oleh *Ralph Merkle* dan *Martin Hellman* in 1978. Disebut juga algoritma *Merkle-Hellman*. Algoritma ini didasarkan pada persoalan 1/0 *Knapsack Problem* yang berbunyi:

Diberikan bobot knapsack adalah M. Diketahui n buah objek yang masing-masing bobotnya adalah w1, w2, …, wn. Tentukan nilai bi sedemikian sehingga

$$M = b1w1 + b2w2 + \ldots + bnwn$$

yang dalam hal ini, bi bernilai 0 atau 1. Jika bi = 1, berarti objek i dimasukkan ke dalam knapsack, sebaliknya jika bi = 0, objek i tidak dimasukkan. Dalam teori algoritma, persoalan knapsack termasuk ke dalam kelompok NP-complete. Persoalan yang termasuk NP-complete tidak dapat dipecahkan dalam orde waktu polinomial. Ide dasar dari algoritma knapsack adalah mengkodekan pesan sebagai rangkaian solusi dari dari persoalan knapsack. Setiap bobot wi di dalam persoalan knapsack merupakan kunci rahasia, sedangkan *bit-bit plainteks* menyatakan bi.

Contoh 1: Misalkan n = 6 dan w1 = 1, w2 = 5, w3 = 6, w4 = 11, w5 = 14, dan w6 = 20.

Plainteks: 111001010110000000011000

Plainteks dibagi menjadi *blok* yang panjangnya n, kemudian setiap *bit* di dalam *blok* dikalikan dengan wi yang berkorepsonden sesuai dengan persamaan (1):

Blok *plainteks* ke-1 : 111001

Kriptogram : $((1 \times 1) + (1 \times 5) + (1 \times 6) + (1 \times 20) = 32$

Blok *plainteks* ke-2 : 010110

Kriptogram : $(1 \times 5) + (1 \times 11) + (1 \times 14) = 30$

Blok *plainteks* ke-3 : 000000

Kriptogram : 0

Blok *plainteks* ke-4 : 011000

Kriptogram : $(1 \times 5) + (1 \times 6) = 11$

Jadi, *Ciphertek*s yang dihasilkan: 32 30 0 11

2.9 *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 *bit plaintext* dan *bit key*. 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).

2.10 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.

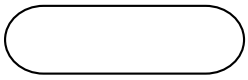
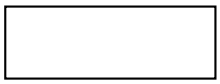
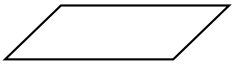
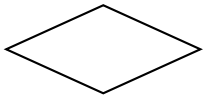
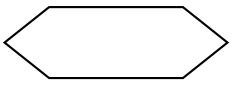
2.11 Pengertian *Flowchart*

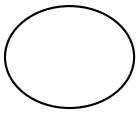
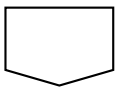
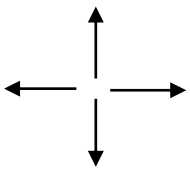
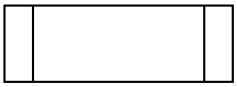
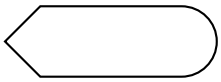
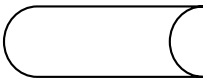
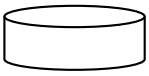
Flowchart adalah sekumpulan simbol-simbol yang menggambarkan rangkaian kegiatan-kegiatan dari data yang akan diproses dalam suatu program dari awal hingga akhir atau suatu bagan yang menggambarkan alir suatu logika

dari data yang akan diproses dalam suatu program dari awal sampai akhir bagan alir terisi dari simbol-simbol yang mewakili fungsi-fungsi langkah program dan garis alir *(flowlines*) menunjukan alir terdiri dari simbol-dimbol yang akan dikerjakana. Tujuan utama pembuatan utama *flowchart* ini adalah untuk menggambarkan suatu tahapan penyelesaian masalah sederhana, rapi dan jelas

Flowchart atau diagram ali merupakan simbol-simbol atau skema yang menunjukkan/menngambarkan rangkaian kegiatan-kegiatan program dari awal hingga akhir. *Flowchart* ini merupakan penggambaran dari urutan langkah-langkah pekerjaan dari suatu algoritma. Adapun simbol *flowchart* dilihat pada tabel sebagai berkut:

Tabel 2.1 Simbol-Simbol *Flowchart*

NO	SIMBOL	FUNGSI
1		**Terminal**, untuk memulai atau mengakhiri suatu program
2		**Proses**, suatu simbol yang menunjukkan setiap pengolahan yang dilakukan
3		***Input-Output***, untuk memasukkan menunjukkan hasil dari suatu proses
4		***Decision***, suatu kondisi yang akan menghasilkan beberapa kemungkinan jawaban atau pilihan
5		***Preparation***, suatu simbol yang menyediakan tempat pengolahan

6		***Connector***, suatu prosedur penghubung yang akan masuk atau keluar melalui simbol ini dalam lembar yang sama
7		***Off-Page Connector***, merupakan simbol masuk atau keluarannya suatu prosedur pada lembaran kertas lainnya
8		**Arus/*Flow***, dari pada prosedur yang dapat dilakukan atas ke bawah dari bawah ke atas, keatas dari kiri ke kanan ataupun dari kanan ke kiri
9		***Predefined Process***, untuk menyatakan sekumpulan langkah proses yang ditulis sebagai prosedur
10		Simbol untuk *output*, yang ditunjukkan ke suatu *device*, seperti printer dan sebagainya
11		Penyimpanan *file* secara sementara
12		Menunjukkan *input* / *outputhardisk* (media penyimpanan)

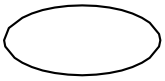
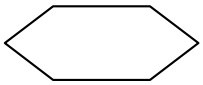
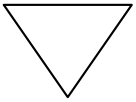
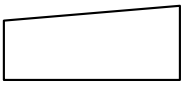
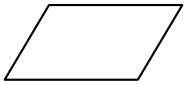
Dalam penulisan *Flowchart* dikenal dua model yaitu sebagai berikut:

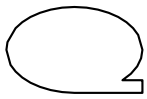
1. Sistem *Flowchart*

   Sistem *Flowchart* yaitu bagan yang memperlihatkan urutan proses dalam sistem dengan menunjukan alat media *input, output* serta jenis media penyimpanan dalam proses pengolahan data.

2. Program *Flowchart*

Program *Flowchart* yaitu bagan yang memperlihatkan urutan intruksi yang digambarkan dengan simbol tertentu untuk memecahkan masalah dengan suatu program

13		Simbol *terminal*, yaitu menyatakan permulaan atau akhir suatu program
14		Simbol *predefined process*, yaitu menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal
15		Simbol *keying operation*, menyatakan segala jenis operasi yang diproses dengan menggunakan suatu mesin yang mempunyai *Keyboard*
16		Simbol *offline-storage*, menunjukkan bahwa data dalam simbol ini akan disimpan ke suatu media tertentu
17		Simbol *manual input*, memasukkan data secara manual dengan menggunakan *online keyboard*
18		Simbol *input/output*, menyatakan proses *input* atau *output* tanpa tergantung jenis peralatannya

19		Simbol magnetic tape, menyatakan input berasal dari pita magnetis atau output disimpan ke dalam pita magnetis
20		Simbol disk storage, menyatakan input berasal dari disk atau output disimpan ke dalam disk
21		Simbol document, mencetak keluaran dalam bentuk dokumen (melalui printer)
22		Simbol punched card, menyatakan input berasal dari kartu atau output ditulis ke kartu.

2.12 *Unified Modeling Language* (UML)

1. Pengenalan *Unified Modeling Language* (UML)

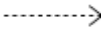
Unified Modelling Language (UML) adalah suatu alat untuk memvisualisasikan dan mendokumentasikan hasil analisis dan desain yang berisi sintak dalam memodelkan sistem secara *visual* (Haviluddin, 2015). Banyak orang yang telah membuat bahasa pemodelan pembangunan perangkat lunak sesuai dengan teknologi pemrograman yang berkembang pada saat itu, misalnya yang sempat berkembang dan digunakan oleh banyak pihak adalah *Data Flow 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).

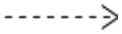
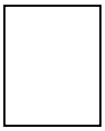
Pada perkembangan teknik pemrograman berorientasi objek, muncullah sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemrograman berorientasi objek, yaitu *Unified Modeling Language* (UML).

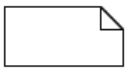
2. *Use Case Diagram*

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 2.2 Simbol *Use Case Diagram*

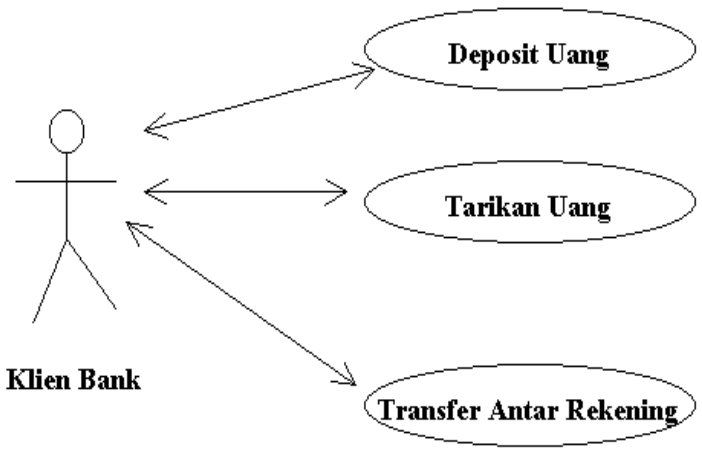
NO	GAMBAR	NAMA	KETERANGAN
1		*Actor*	Menspesifikasikan himpuan peran yang pengguna mainkan ketika berinteraksi dengan *use case*.
2		*Dependency*	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri *(independent)* akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (*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 2.4. Contoh *Use Case Diagram*
Sumber : (Haviluddin, 2015)

3. *Activity Diagram*

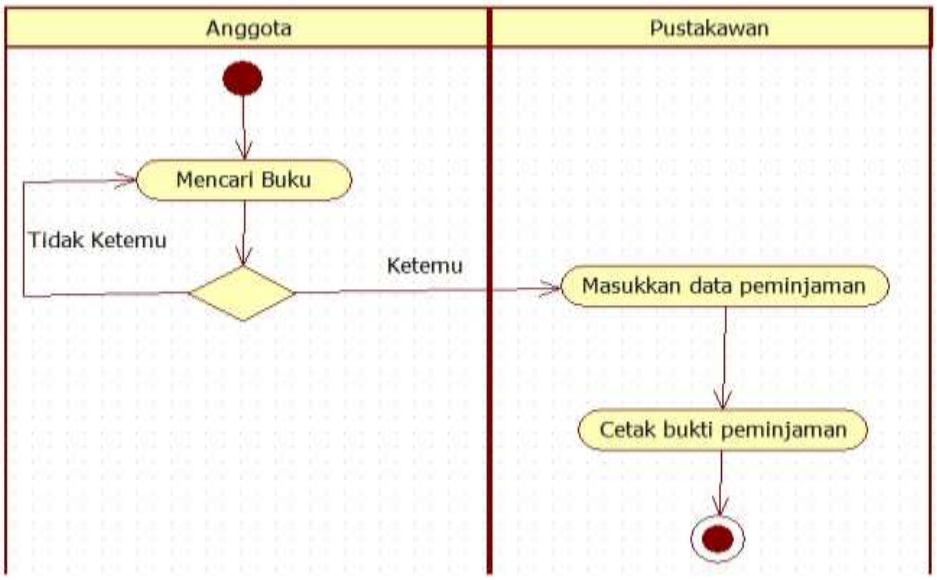
Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau *menu* yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor,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

Tabel 2.3. 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, 2015)

Contoh Activity Diagram :



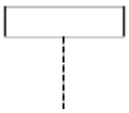
Gambar 2.5. Contoh *Activity Diagram*

Sumber : (Gellysa Urva, 2015)

4. *Sequence Diagram*

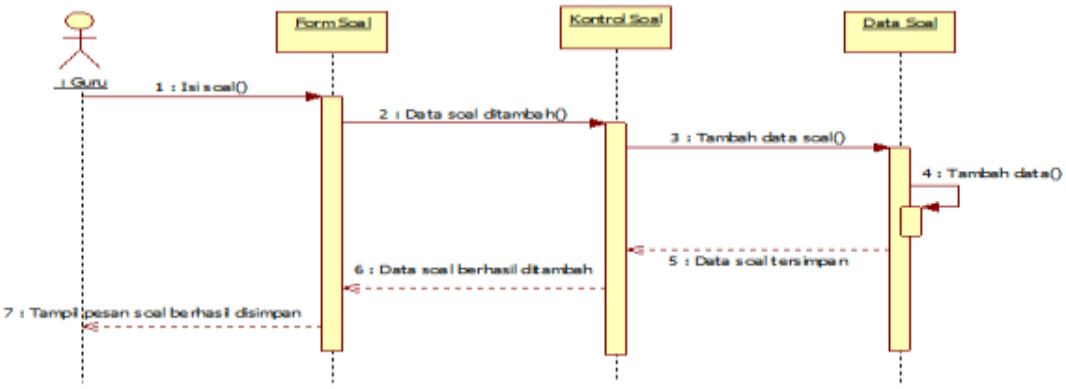
Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambar diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang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

Tabel 2.4. Simbol *Sequence Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		*LifeLine*	Objek *entity*, antarmuka yang saling berinteraksi.
2		*Message*	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi.
3		*Message*	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi.

Sumber : (Gellysa Urva, 2015)

Contoh Squence Diagram :

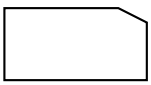


Gambar 2.6. Contoh *Squence Diagram*
Sumber : (Gellysa Urva, 2015)

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.

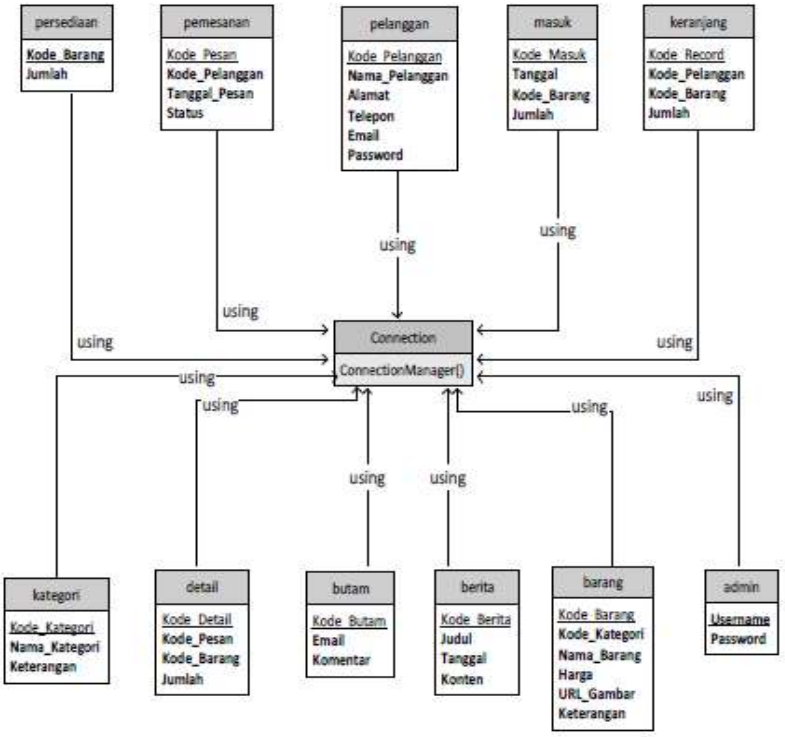
Tabel 2.5. Simbol *Class Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		*Note*	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

2		*dependency*	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri akan mempengaruhi elemen yang bergantung padanya
3		*Extend*	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 2.7. Contoh *Class Diagram*

Sumber : (Gellysa Urva, 2015)

2.13 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 (Melina, 2014).

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.

2.14 Pengertian *Message-Digest* 5 (MD5)

MD5 merupakan singkatan dari *message-digest algortihm 5*, adalah fungsi *hash* (prosedur terdefinisi atau fungsi matematika yang mengubah *variabel* dari suatu data yang berukuran besar menjadi lebih sederhana*) kriptografi* yang digunakan secara luas dengan *hash value* 128-bit. MD5 dimanfaatkan dalam berbagai aplikasi keamanan, dan umumnya digunakan untuk menguji integritas sebuah *file. Enkripsi* menggunakan MD5 mendominasi sebagian besaraplikasi PHP. *Enkripsi* MD5 dianggap strong karena enkripsi yang dihasilkannya bersifat '*one way hash*'. Berapapun string yang di enkripsi hasilnya tetap sepanjang 32 karakter.

Hash MD5 sepanjang 128-bit (16 byte), yang dikenal juga sebagai ringkasan pesan, secara tipikal ditampilkan dalam bilangan *heksadesimal* 32-digit.berikut ini merupakan contoh pesan ASCII sepanjang *43-byte* sebgai masukan *hash* MD5 terkait:

MD5 (“*The quick brown fox jumps over the lazy dog*”)

=9e107d9d372bb6826bd81d3542a419d6

Bahkan perubahan yang kecil pada pesan akan (dengan probabilitas lebih) menghasilkan hash yang benar-benar berbeda, misalnya pada kata “*dog*”, huruf d diganti menjadi c:

MD5 *(‘The quick brown fox jumps oer the lazy cog”)*

=1055d3e698d289f2af8663725127bd4b

Hash dari panjang-nol ialah:

MD5(“”)

=d41d8cd90f00b204e9800998ecf8427e

Ringkasan MD5 digunakan secara luas dalam dunia perangkat lunak untuk menyediakan semacam jaminan bahwa berkas yang diambil (download) belum terdapat perubahan. Seorang pengguna dapat membandingkan MD5 yang dipublikasi dengan *checksum* dari berkas yang diambil. Dengan asumsi bahwa *checksum* yang dipublikasikan dapat dipercaya akan keasliannya, seorang pengguna dapat secara yakin bahwa berkas tersebut adalah berkas yang sama dengan berkas yang dirilis oleh para *developer*, jaminan perlindungan dari *trojan horse* dan virus komputer yang ditambahkan pada perangkat lunak. Bagaimanapun juga, seringkali kasus yang terjadi bahwa checksum yang dipublikasikan tidak dapat dipercaya (sebagai contoh, *checksum* didapat dari *chennel* atau lokasi yang sama dengan tempat mengambil berkas), dalam hal ini MD5 akan dikenali bekas yang didownload tidak sempurna, cacat atau tidak lengkap.

Untuk aplikasi penguujian integritas sebuah *file* atau lebih dikenal dengan istilah MD5 *Checksum*, dapat menggnakan aplikasi dekstop atau aplikasi bebasis web MD5 *Checksum* seperti"MD5 *Checksum Verifier*" dan sebagainya. *Software* semacam ini akan menghasilkan kode MD5 dari *file* yang diuji integritasnya. Selanjutnya kode MD5 ini akan digunakan untuk menguji apakah *file* tersebut memiliki integritas ataukah tidak. Artinya jika *file* akan diberikan atau dikirimkan atau diunduh, penerima dapat mencocokkan dengan yang diterima apakah ukuran, struktur, dan jenis *file* sesuai dengan diberikan oleh pembuat *file*. Contohnya Checksumnya , jika diperikasa (*divalidasi*) dengan *tool* seperti MD5 *Checksuum Verifier*, dinyatakan *valid* atau sama dengan *file* yang diuji, maka dikatakan dile tersebut tak mengalami perubahan dari pengirim hingga ke tangan anda. (perubahan bisa terjadi karena virus dan sebagainya). Pengujian semacam ini ditunjukan untuk memastikan suatu *file* tidak *corrupt* atau mungkin terinveksi, baik itu karena *virus, malware*, atau *injeksi software* berbahaya lainnya. (Saipul Bahri, 2018

2.15 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*.

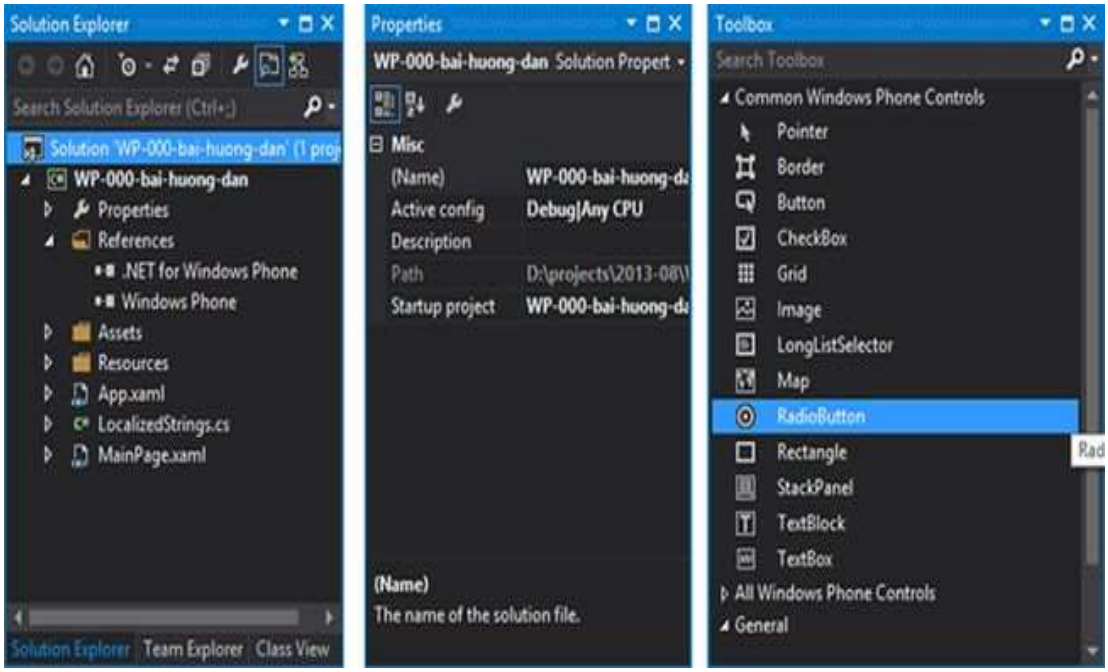
1. Komponen kerja

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 2.8. Tampilan *Toolbox*
Sumber : (Safik, 2015)

Table 2.6. *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)

2.16 *American Standard Code for Information Interchange* (ASCII)

ASCII merupakan kepanjangan dari *American Standard Code for Information Interchange* (ASCII), dan pengertian dari ASCII sendiri adalah suatu standar internasional dalam kode huruf dan simbol seperti *Hex* dan *Unicode* tetapi ASCII lebih bersifat universal.

Tabel 2.7 Karakter Tabel ASCII

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	

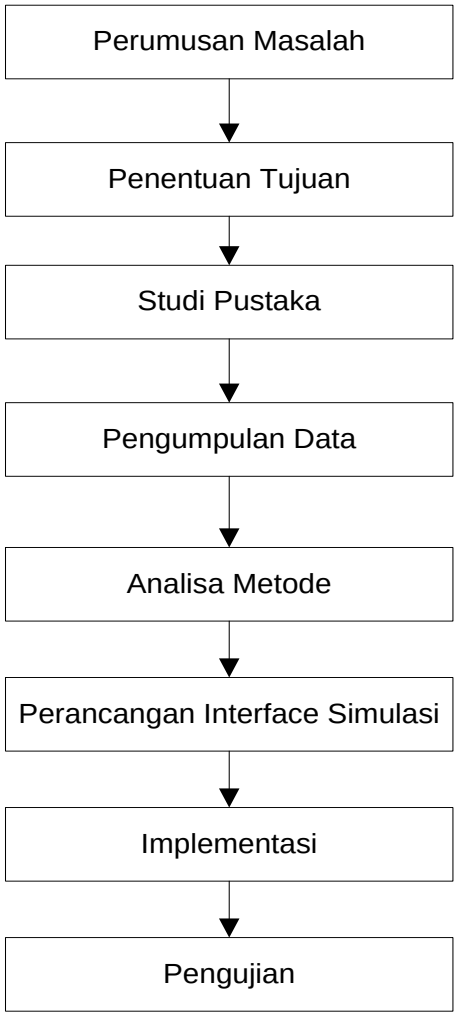
Sumber: https://www.ascii-code.com/

BAB III

ANALISIS DAN PERANCANGAN SISTEM

3.1 Tahapan Penelitian

Adapun tahapan penelitian yang dilakukan oleh penulis ini dengan judul Penerapan MD5 Pada Verifikasi Dan Validasi Keaslian Data 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. Wawancara (*Interview*).

   Wawancara ini dilakukan dengan cara mengadakan komunikasi langsung dengan dosen pengampu mata kuliah kemanan data di Universitas Pembangunan Pancabudi Medan yang dapat memberikan informasi dan data-data yang diperoleh mengenai kemanaan data.

2. Pengamatan (*Observation*)

   Penulis melakukan pengamatan langsung pada setiap pengumpulan data .

3. 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

Sistem Informasi Keamanan data yang sedang berjalan adalah sebagai berikut pada saat proses Keamanan data, masih melakukan dengan cara manual untuk proses keamanan data, hal ini sering menyebabkan kerusakan pada data

yang sering dibawa pada saat pengumpulan data. Data yang dikumpulkan masih bersifat tidak rahasia sehingga keamanan dari file tersebut tidak baik.

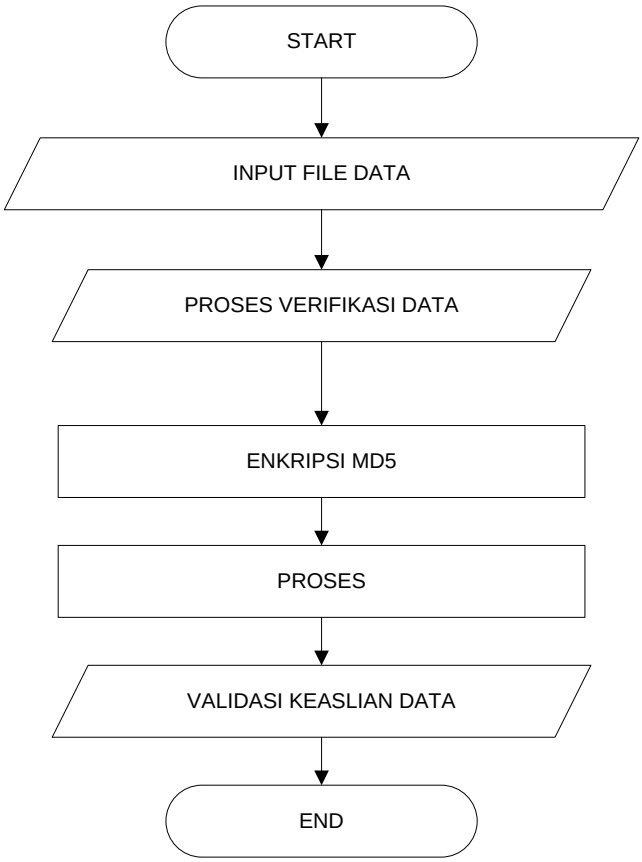
Berdasarkan analisis sistem yang sedang berjalan, maka dapat disimpulkan sebagai berikut:

1. Kemanan Data keamanan data tidak ada.
2. Data keamanan data dapat rusak dan hilang sehingga terjadi penumpukan dan kerusakan data.

3.4 Analisis Sistem Yang Diusulkan

Analisis prosedural atau proses sistem memberikan gambaran tentang sistem yang saat ini berjalan. Analisis sistem bertujuan untuk mengetahui lebih jelas bagaimana cara kerja sistem tersebut, sehingga kelebihan dan

kekurangan sistem dapat diketahui. Prosedur itu sendiri merupakan urutan kegiatan yang paling tepat dari tahapan – tahapan yang menerangkan mengenai proses apa yang dikerjakan, siapa yang mengerjakan proses tersebut bagaimana proses tersebut dapat dikerjakan dan apa saja yang terlibat. Berikut untuk penjelasan yang lebih jelasnya dalam *flowchart* dibawah ini :



Gambar 3.2. *Flowchart* Sistem Yang Diusukan

3.5 Analisis Kebutuhan *Non-Fungsional*

Analisis kebutuhan *non fungsional* merupakan analisis yang dibutuhkan untuk menentukan spesifikasi kebutuhan sistem. Spesifikasi ini juga meliputi elemen atau komponen – komponen apa saja yang dibutuhkan untuk sistem yang akan dibangun sampai dengan sistem tersebut diimplementasikan. Analisis kebutuhan ini juga menentukan spesifikasi masukkan yang diperlukan sistem, keluaran yang akan dihasilkan sistem dan proses yang dibutuhkan untuk mengolah masukkan sehingga menghasilkan suatu keluaran yang diinginkan.

3.5.1 Analisis Perangkat Keras (*Hardware*)

Perangkat keras minimum yang digunakan untuk membangun Perancangan Verifikasi Data ini adalah

1. Processor berkecepatan 3.0 Ghz
2. RAM 4 Gb
3. Hardisk minimal 10 Gb untuk menyimpan data
4. Keyboard dan Mouse
5. Monitor 14.

3.5.2 Analisis Perangkat Lunak (*Software*)

Untuk mendukung dalam penyimpanan informasi, dibutuhkan suatu fasilitas yang memadai. Yaitu berupa perangkat lunak (*software*) yang dirancang untuk memudahkan dalam pembangunan dan menjalankan sisten nantinya. Adapun perangkat lunak yang digunakan adalah sebagai berikut :

1. Microsoft Windows 7 , Windows 8 sebagai sistem operasi
2. Visual Studio 2010

3.6 Perhitungan MD5

Adapun proses perhitungan manual pada proses MD5 nya adalah sebagai berikut:

Contoh:

Text di Conver menjadi bilangan *Hexdecimal* adalah sebagai berikut:

07 f1 72 34 28 ab

Lalu hitung pergantian hex nya yaitu ditambah 8 dan menggunakan hex 16 bit ([0-9] [a-f]) dan hex yang diambil adalah hex yang paling depan (sebelah kiri) lalu mari kita ganti hex nya sesuai dengan interval di atas

hex 68 diganti menjadi E8

cara perhitungan :

6 + 8 = E

7,8,9,a,b,c,d,e

hex 5F diganti menjadi DF

cara perhitungan :

5 + 8 = d

6,7,8,9,a,b,c,d

hex 89 diganti menjadi 09

cara perhitungan :

8 + 8 = 0

9,a,b,c,d,e,f,0

hex 7E diganti menjadi FE

cara perhitungan :

7 + 8 = f

8,9,a,b,c,d,e,f

hex F5 diganti menjadi 75

cara perhitungan :

F + 8 = 7

0,1,2,3,4,5,6,7

hex 1E diganti menjadi 9E

cara perhitungan :

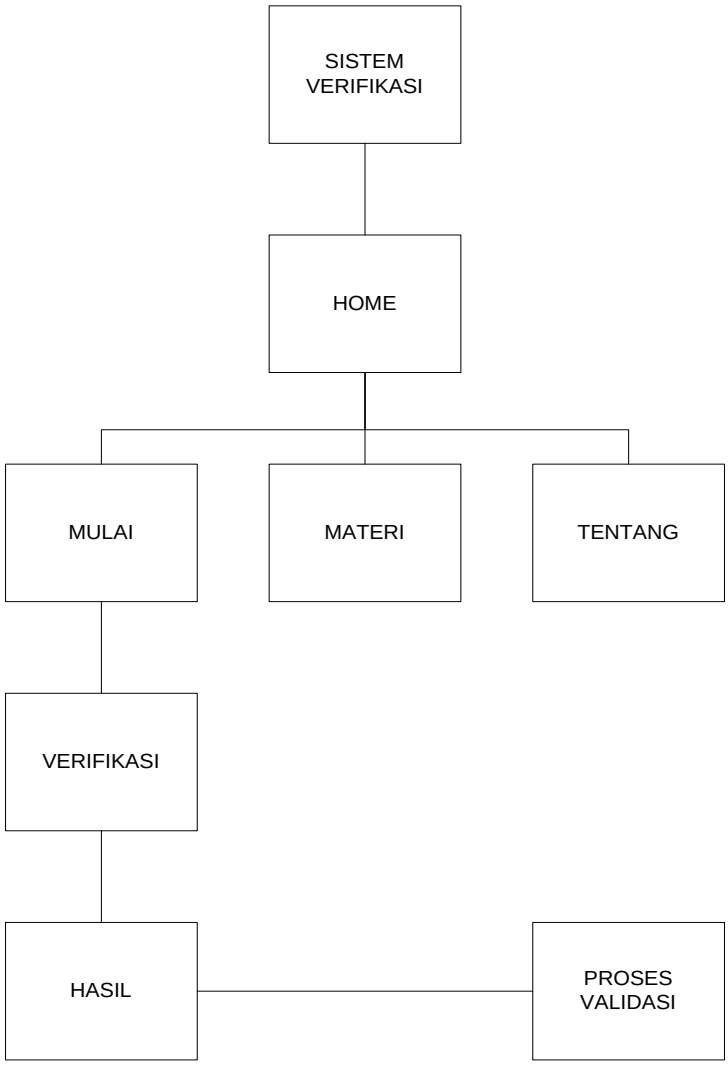
1 + 8 = 9

2,3,4,5,6,7,8,9

Maka hasil dari Proses MD5 adalah : E8DF09FE759E

3.7 Perancangan *Flowmap*

Sebelum dilakukannya proses perancangan tampilan pada sistem verifikasi dan keaslian data, maka terlebih dahulu dilakukan *flowmap* pada perancangan sistem, adapun *flowmap* ada Sistem verifikasi dan keaslian data adalah sebagai berikut:



Gambar 3.4 Rancangan *FlowMap* Sistem *verifikasi* dan keaslian data

3.8 Perancangan Tampilan

Perancangan merupakan bagian yang paling penting dalam merancang sistem. Adapun bentuk Sistem verifikasi dan keaslian data adalah sebagai berikut:

2.8.1 Rancangan Halaman Judul

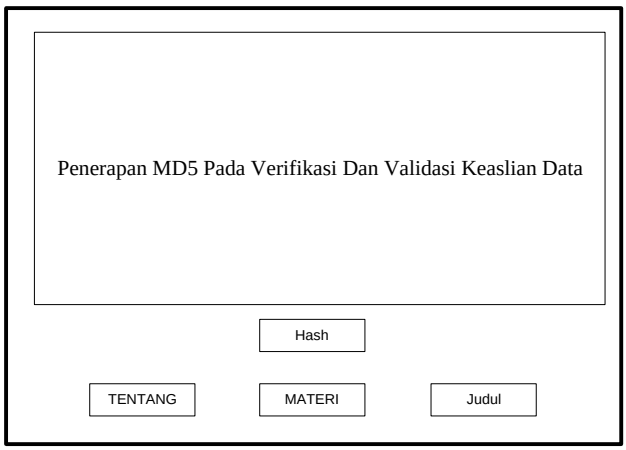
Halaman judul merupakan halaman yang pertama muncul pada saat program dijalankan

Gambar 3.5 Rancangan Halaman Judul

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

2.8.2 Rancangan Halaman Menu Utama

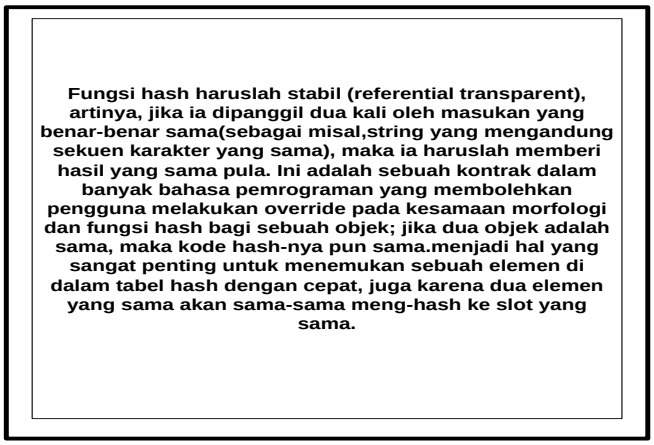
Form ini berisi tombol-tombol seperti menu Materi, Hash, Judul, tentang, dan Keluar.



Gambar 3.6 Rancangan Halaman Menu Utama

2.8.3 Rancangan Halaman Materi

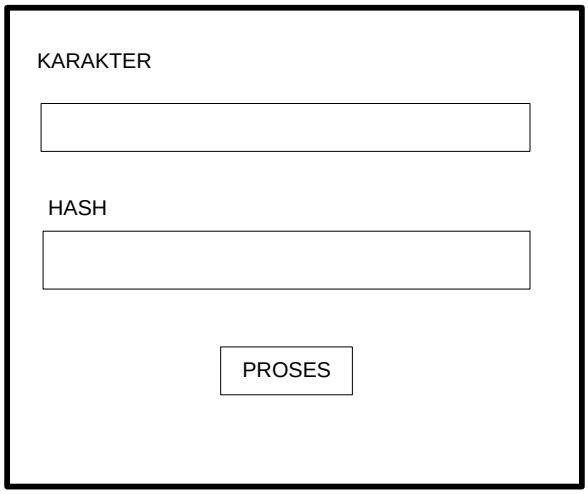
Form ini digunakan untuk menjelaskan cara kerja penyandian, dimulai dari plaintext kemudian kunci yang dikonversikan dalam bentuk angka. Setelah itu dilakukan proses penjumlahan dan jika hasil penjumlahan maka akan dikurangi 6 lalu hasilnya akan dikembalikan lagi ke dalam bentuk huruf.



Gambar 3.7 Rancangan Halaman Materi

2.8.4 Rancangan Halaman Hash

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



Gambar 3.8 Rancangan Halaman Hash

BAB IV

HASIL DAN PEMBAHASAN

4.1 Implementasi

Menu yang terdapat didalam aplikasi ada berupa seorang pengguna. Sebelum mengaplikasikan aplikasi pengguna harus membuka program Aplikasinya, Menu yang dapat diaplikasikan oleh pengguna adalah *Materi, Tentang, Judul*. Sedangkan pengguna mengaplikasikan menu Mulai untuk melakukan proses enkripsi MD5.

4.1.1 Spesifikasi Sistem

Analisis kebutuhan sistem merupakan analisis yang dibutuhkan untuk menentukan spesifikasi kebutuhan sistem. Spesifikasi ini juga meliputi elemen atau komponen – komponen apa saja yang dibutuhkan untuk sistem yang akan dibangun sampai dengan sistem tersebut diimplementasikan. Analisis kebutuhan ini juga menentukan spesifikasi masukkan yang diperlukan sistem, keluaran yang akan dihasilkan sistem dan proses yang dibutuhkan untuk mengolah masukkan sehingga menghasilkan suatu keluaran yang diinginkan.

1. Analisis Perangkat Keras (*Hardware*)

Perangkat keras minimum yang digunakan untuk membangun Perancangan Verifikasi Data ini adalah

a. Processor Berkecepatan 3.0 Ghz

b. RAM 4 Gb

c. Hardisk minimal 10 Gb untuk menyimpan data

d. Keyboard dan Mouse

e. Monitor 14 inch.

2. Analisis Perangkat Lunak (*Software*)

Untuk mendukung dalam penyimpanan informasi, dibutuhkan suatu fasilitas yang memadai. Yaitu berupa perangkat lunak (software) yang dirancang untuk memudahkan dalam pembangunan dan menjalankan sisten nantinya. Adapun perangkat lunak yang digunakan adalah sebagai berikut :

a. Microsoft Windows 8 , Windows 8 sebagai sistem operasi

b. Visual Studio 2010, Sebagai Perancangan Program Aplikasi.

4.2 Hasil Rancangan Sistem

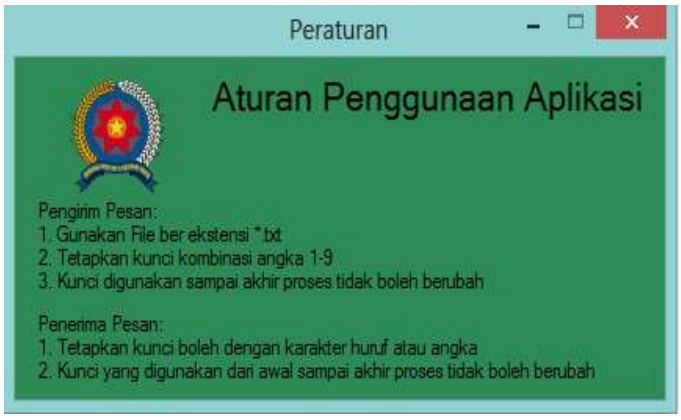
4.2.1 Tampilan Awal/ Home

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 materi yang akan mengarahkan pengguna ke form yang menjelaskan tata cara penggunaan dari aplikasi ini.

Gambar 4.1 Tampilan Awal/ Home

4.2.2 Tampilan Aturan Penggunaan Aplikasi

Tampilan aturan penggunaan aplikasi merupakan tampilan halaman atau form yang berisi tentang tata cara penggunaan aplikasi yang dijalankan. Pada halaman tersebut dijelaskan apa-apa saja yang menjadi kewajiban bagi pengirim dan penerima saat menjalankan simulasi algoritma MD5.



Gambar 4.2 Tampilan Aturan Penggunaan Aplikasi

4.2.3 Tampilan Halaman Pengirim Pesan

Tampilan berikut merupakan tampilan pengiriman pesan pada aplikasi ini. algoritma MD5 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 men*dekripsi* pesan tanpa perlu mengetahui kunci yang lainnya

Gambar 4.3 Tampilan Halaman Pengirim Pesan

4.3 Kesimpulan Dan Hasil Pengujian Sistem

Untuk dapat menggunakan aplikasi ini dengan baik, dibutuhkan seperangkat komputer dengan spesifikasi minimal. kesimpulan dan hasil pengujian adalah metode pengujian perangkat lunak yang menguji fungsionalitas aplikasi yang bertentangan dengan struktur internal atau kerja. Metode uji dapat diterapkan pada semua tingkat pengujian perangkat lunak: unit, integrasi, fungsional, sistem dan penerimaan.

Tabel 4.1. Tabel Pengujian

No	Rancangan Proses	Hasil Yang Diharapkan	Hasil	Keterangan
1	Index	Halaman Index (Awal)	Sesuai	-
2	Input Data	Halaman Data Siswa	Sesuai	-
3	View Data	Halaman Tampil Data	Sesuai	-
4	View Berkas	Halaman Hasil MD5	Sesuai	-

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

Tabel 4.2. Kesimpulan Pengujian Alpha

Nama fungsi	Hasil
Index	Fungsi berjalan dengan baik
Input Data	Fungsi berjalan dengan baik
View Data	Fungsi berjalan dengan baik
View Berkas	Fungsi berjalan dengan baik

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan pembahasan dalam perancangan Sistem Verifikasi dan Validasi Keaslian Data Menggunakan MD5, maka dapat diambil kesimpulan sebagai berikut :

1. Perangkat lunak ini dirancang untuk mengamankan file data text pada proses enkripsi menggunakan metode MD5.
2. Penggunaan metode MD5 sangat baik digunakan untuk proses pengamanan file.
3. Penggunaan kunci sulit di tebak dikarenakan menggunakan hexdecimal to binary.

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

Al Bahra. (2005). Analisis dan Desain Sistem Informasi. Edisi Pertama Tangerang : Penerbit Graha Ilmu. https://id.scribd.com.

Alfha Vionita, Dyah Purboningsih. (2016). Pengguna Metode Enkripsi Vigenere

Barus, S., Sitorus, V. M., Napitupulu, D., Mesran, M., & Supiyandi, S. (2018). Sistem Pendukung Keputusan Pengangkatan Guru Tetap Menerapkan Metode Weight Aggregated Sum Product Assesment (WASPAS). Jurnal Media Informatika Budidarma, 2(2).

Batubara, Supina. "Analisis perbandingan metode fuzzy mamdani dan fuzzy sugeno untuk penentuan kualitas cor beton instan." IT Journal Research and Development 2.1 z, Supina, Sri Wahyuni, and Eko Hariyanto. "Penerapan Metode Certainty Factor Pada Sistem Pakar Diagnosa Penyakit Dalam." Seminar Nasional Royal (SENAR). Vol. 1. No. 1. 2018.

dan MD5 dalam Proses Pengamanan Pesan. (5).

Gilang Gumira P.U.K, Ernawati, Aan Erlanshari. Implementasi Metode Advanced Encryption Standard (AES) dan Message Digest 5 (MD5) Pada Enkripsi Dokumen. *https://ejournal.unib.ac.id .*

Hariyanto, E., & Rahim, R. (2016). Arnold's cat map algorithm in digital image encryption. International Journal of Science and Research (IJSR), 5(10), 1363-1365.

Hastanti, R. P., & Purnama, B. E. (2015). Sistem Penjualan Berbasis Web (E-Commerce) Pada Tata Distro Kabupaten Pacitan. Bianglala Informatika, 3(2).

Hidayatus Sibayan. (2016). Implementasi Basis Data Dengan Algoritma MD5 (Message Digest Algorithm 5) dan Vigenere Cipher. (3). https://ojs.unsiq.ac.id/index.php/ppkm/article/view/412/242.

https://docplayer.info/storage.

Inayatullah (2007). Analisa Penerapan Algoritma MD5 untuk Pengamanan Password, 1 (2). http://eprints.mdp.ac.id/553.

Khairul, K., IlhamiArsyah, U., Wijaya, R. F., & Utomo, R. B. (2018, September). Implementasi Augmented Reality Sebagai Media Promosi Penjualan Rumah. In Seminar Nasional Royal (Senar) (Vol. 1, No. 1, pp. 429-434).

Kurniawan, H. (2018). Pengenalan Struktur Baru untuk Web Mining dan Personalisasi Halaman Web. Jurnal Teknik dan Informatika, 5(2), 13-19

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

Munawir, M., & Ardiansyah, A. (2017). Decision Support System Pemilihan Karyawan Berprestasi Dengan Pendekatan Analisa Gap Profile matching Di Kantor Perwakilan Bank Indonesia Provinsi Aceh. Jurnal JTIK (Jurnal Teknologi Informasi dan Komunikasi), 1(1), 7-14.

Nugroho, B. (2013). Dasar pemrograman web PHP-MySQL dengan Dreamweaver. Yogyakarta: Gava Media.

Precilia, D. P., & Izzuddin, A. (2015). Aplikasi Tanda Tangan Digital (Digital Signature) Menggunakan Algoritma Message Digest 5 (MD5). Energy, 5(1), 14-19.

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

Rahim, R., Aryza, S., Wibowo, P., Harahap, A. K. Z., Suleman, A. R., Sihombing, E. E., ... & Agustina, I. (2018). Prototype file transfer protocol application for LAN and Wi-Fi communication. Int. J. Eng. Technol., 7(2.13), 345-347.

Rahim, R., Supiyandi, S., Siahaan, A. P. U., Listyorini, T., Utomo, A. P., Triyanto, W. A., ... & Khairunnisa, K. (2018, June). TOPSIS Method Application for Decision Support System in Internal Control for Selecting Best Employees. In Journal of Physics: Conference Series (Vol. 1028, No. 1, p. 012052). IOP Publishing.

Rosa, A. S., & Shalahuddin, M. (2015). Rekayasa Perangkat Lunak Terstruktur dan Beroriented Objek. Informatika Bandung.

Saipul Bahri, Diana, Susan Dian PS. (2012). Studi Implementasi Pengamanan Basis Data Menggunakan Metode Enkripsi MD5. Jurnal ilmiah vol.10. 5-8. http://eprints.binadarma.ac.id/258/1.

Sari, R. D., Supiyandi, A. P. U., Siahaan, M. M., & Ginting, R. B. (2017). A Review of IP and MAC Address Filtering in Wireless Network Security. Int. J. Sci. Res. Sci. Technol, 3(6), 470-473.

Siahaan, A. P. U., Aryza, S., Nasution, M. D. T. P., Napitupulu, D., Wijaya, R. F., & Arisandi, D. (2018). Effect of matrix size in affecting noise reduction level of filtering.

Siahaan, MD Lesmana, Melva Sari Panjaitan, and Andysah Putera Utama Siahaan. "MikroTik bandwidth management to gain the users prosperity prevalent." Int. J. Eng. Trends Technol 42.5 (2016): 218-222.

Sidik, A. P. (2018). Algoritma RSA dan Elgamal sebagai Algoritma Tambahan untuk Mengatasi Kelemahan Algoritma One Time Pad pada Skema Three Pass Protocol.

Tasril, V. (2018). Sistem Pendukung Keputusan Pemilihan Penerimaan Beasiswa Berprestasi Menggunakan Metode Elimination Et Choix Traduisant La Realite. INTECOMS: Journal of Information Technology and Computer Science, 1(1), 100-109.