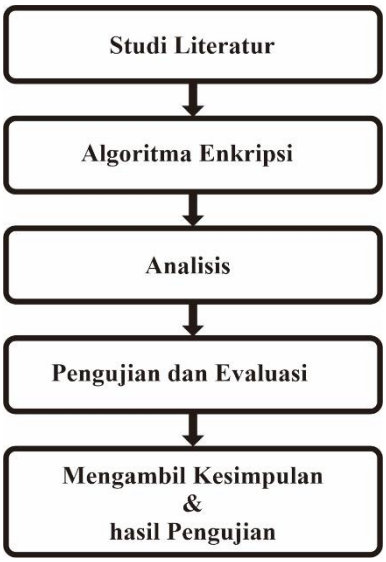
BAB III
METODOLOGI PENELITIAN

3.1 Alur Penelitian

Tahap penelitian adalah bagian yang terstruktur dalam melaksanakan penelitian. Dalam penyusunan tugas akhir ini, langkah-langkah analisis secara umum dapat diuraikan sebagai berikut pada Gambar 3.1



Gambar 3. 1 Alur Penelitian

3.2 Studi Literatur

Melakukan studi literatur guna menghimpun informasi yang diperlukan. Informasi tersebut diperoleh melalui sumber-sumber seperti buku-buku di perpustakaan, artikel, jurnal, publikasi ilmiah, dokumen, dan sumber-sumber elektronik yang ditemukan melalui situs-situs internet.

3.3 Analisis

Dalam penelitian ini penulis menggunakan metode analisis deskriptif untuk menganalisa data-data yang diperoleh.

3.3.1 Analisa Kebutuhan Pengguna

Berdasarkan pengamatan dalam penelitian ini maka dibutuhkan beberapa perangkat pendukung dalam pelaksanaannya untuk men simulasikan daripada kinerja enkripsi yang diterapkan.

3.3.2 Analisa Kebutuhan Perangkat

diantaranya adalah :

Analisa Perangkat Lunak

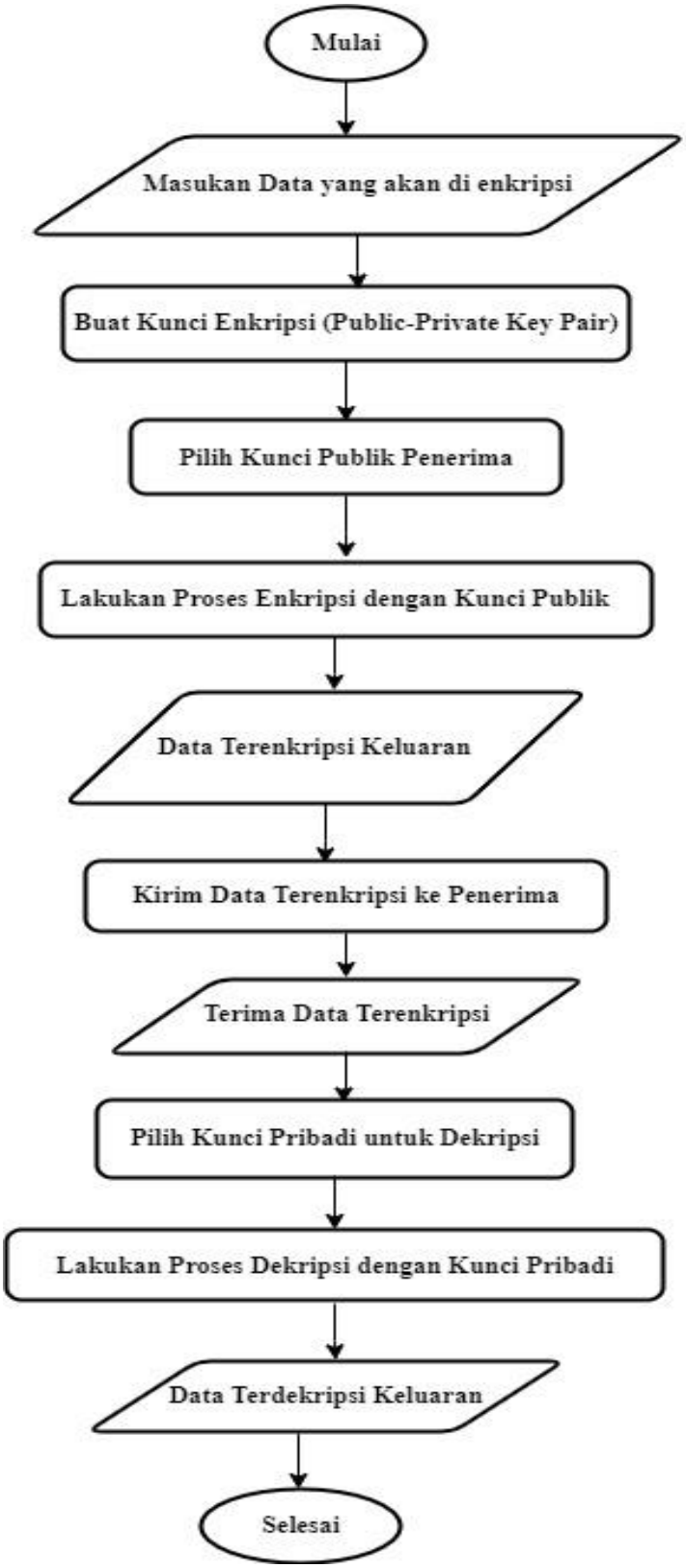
a) Sistem Operasi Linux Ubuntu 22.04.3 LTS
b) Oracle VM VirtualBox
c) *GNU Privacy Guard* Vers 2.4.3

Analisa Perangkat Keras

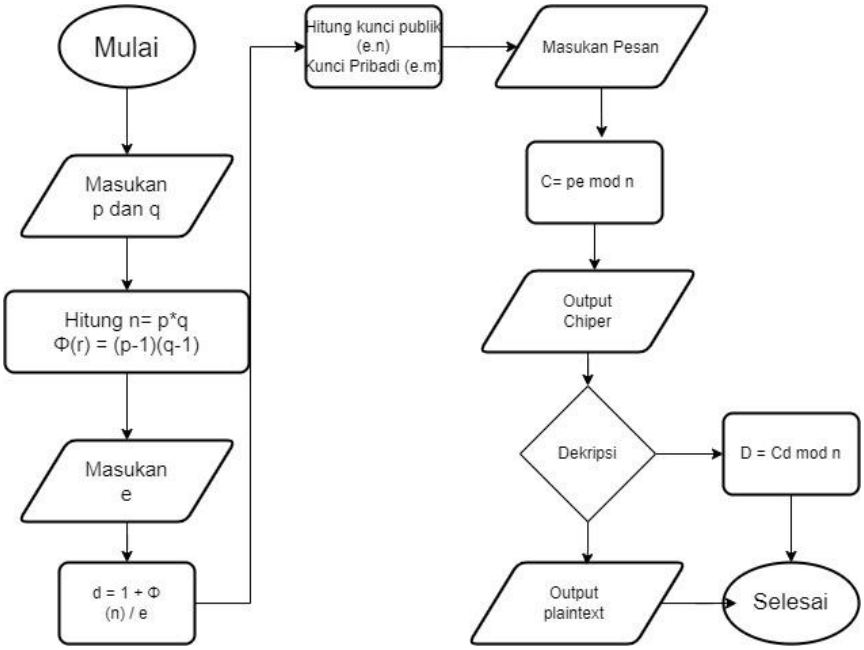
a) Processor Intel Core i5-7th Gen
b) Ram 8 GB
c) VGA Nvidia Geforce 940 mx

3.4 Flowchart

Pada Proses ini merupakan perancangan alur yang akan digunakan pada penerapan sistem enkripsi, yaitu dengan tahapan Flowchart. *Flowchart* merupakan representasi grafis dari alur kerja atau proses dalam bentuk diagram. Flowchart digunakan untuk menggambarkan langkah-langkah, keputusan, percabangan, dan aliran informasi dalam suatu proses. Flowchart membantu secara visual memahami proses tersebut dengan jelas dan memudahkan untuk dianalisis, ditingkatkan, atau dijelaskan. Adapun *flowchart* yang ditunjukan pada Gambar 3.2



Gambar 3. 2 Flowchart enkripsi Pada Gnupg



Gambar 3. 3 Flowchart Algoritma Pada Enkripsi RSA

3.5 RSA (Rivest-Shamir-Adleman)

Langkah pertama dalam proses enkripsi menggunakan algoritma RSA adalah mengubah *plaintext* "INFORMATIKA" ke dalam format ASCII. Setiap karakter dalam *plaintext* akan diwakili oleh nilai ASCII yang sesuai :

Tabel 3. 1 Mengubah *Plaintext* Ke ASCII

Text (Karakter)	**I**	**N**	**F**	**O**	**R**	**M**	**A**	**T**	**I**	**K**	**A**
ASCII (Heksa)	49	4E	46	4F	52	4D	41	54	49	4B	41
ASCII (Desimal)	73	78	70	79	82	77	65	84	73	75	65

Data teks yang tersaji dalam bentuk ASCII desimal selanjutnya dipisah ke dalam segmen-segmen tiga angka seperti yang berikut:

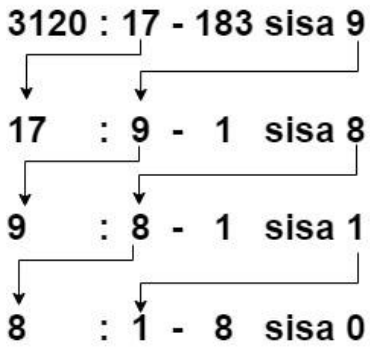
m^1= 737 m^2= 870 m^3= 798 m^4= 277

m^5= 658 m^6= 473 m^7= 756 m^8= 005

Selanjutnya Pilih dua bilangan prima besar secara acak, p dan q. Misalkan dipilih p=61 dan q=53 (Keduanya prima), maka dapat dihitung

$$n = p \times q = 3233$$
$$\Phi(r) = (p-1)(q-1) = 3120$$

Dipilih kunci publik e = 17 (yang relatif prima dengan 3120 karena pembagi bersama terbesarnya adalah 1). Bahwa 17 relatif prima terhadap 3120 dapat dibuktikan dengan mencari nilai gcd (17,3120) melalui algoritma Euclid seperti berikut :



Ketika sisa dari pembagian menjadi 0, maka bilangan yang digunakan terakhir sebelum sisa menjadi 0 adalah FPB dari 17 dan 3120, yaitu 1. Dengan demikian, hasil perhitungan manual menggunakan algoritma Euclidean menunjukkan bahwa FPB dari bilangan prima 17 dan 3120 adalah 1. Ini menandakan bahwa bilangan prima 17 relatif prima dengan 3120.

Maka pembagian menjadi segmen-segmen 3 angka diperoleh enkripsi setiap blok diperoleh menggunakan kunci publik 17 berikut :

Tabel 3. 2 Segmen Enkripsi

$C^1 = 737^{17} \bmod 3233 = 2019$	$C^5 = 658^{17} \bmod 3233 = 2536$
$C^2 = 870^{17} \bmod 3233 = 2839$	$C^6 = 473^{17} \bmod 3233 = 1626$
$C^3 = 798^{17} \bmod 3233 = 2102$	$C^7 = 756^{17} \bmod 3233 = 235$
$C^4 = 277^{17} \bmod 3233 = 2219$	$C^8 = 005^{17} \bmod 3233 = 517$

Dengan melakukan proses enkripsi maka dihasilkan *chipertext*

C = 2019 2839 2102 2219 2536 1626 235 517

Proses dekripsi RSA melibatkan penggunaan kunci privat yang sesuai untuk mengonversi *ciphertext* (teks terenkripsi) kembali menjadi *plaintext* (teks asli).
Rumus dekripsi RSA adalah:

$$m = c^d \ mod \ n$$

Proses dekripsi dilakukan dengan menggunakan kunci rahasia d= 366

Tabel 3. 3 Segmen Dekripsi

$m^1 = 2019^{366}$ mod 3233 = 737	$C^5 = 2536^{366}$ mod 3233 = 658
$m^2 = 2839^{366}$mod 3233 = 870	$C^6 = 1626^{366}$ mod 3233 = 473
$m^3 = 2102^{366}$mod 3233 = 798	$C^7 = 235^{366}$ mod 3233 = 756
$m^4 = 2219^{366}$mod 3233 = 277	$C^8 = 517^{366}$ mod 3233 = 005

Dengan melakukan proses dekripsi maka dihasilkan ASCII

D = 737 870 798 277 658 473 756 005

Dan dihasilkan proses dekripsi berupa *plaintext* menjadi

P = INFORMATIKA

3.6 Cara Kerja GnuPG

GnuPG menawarkan berbagai layanan dalam menerapkan standar *OpenPGP*, seperti menjaga integritas pesan dan data *file* melalui teknologi tanda tangan digital, enkripsi, kompresi, dan konversi Radix-64. Selain itu, GnuPG juga menyediakan layanan untuk manajemen dan sertifikasi kunci. Dengan menggabungkan fitur-fitur terbaik dari kriptografi konvensional dan kriptografi kunci publik, GPG menjadi sistem kriptografi hibrida yang kuat.

GPG menerapkan proses enkripsi dengan memanfaatkan kriptografi kunci publik (asimetris) dan sebuah sistem yang mengintegrasikan kunci publik dengan identitas pengguna. Enkripsi dalam GPG melibatkan penggunaan algoritma enkripsi kunci asimetris yang melibatkan pasangan kunci publik dan kunci privat. Saat mengirim pesan, pengirim menggunakan kunci publik penerima untuk mengenkripsi kunci rahasia yang nantinya digunakan dalam algoritma enkripsi simetris.

Adapun langkah-langkah dalam penerapan enrkripsi menjaga kerahasiaan data proses pengiriman pesan melalui enkripsi GnuPG sebagai berikut :

a) Pengirim membuat pesan.
b) Pengirim membangkitkan sebuah bilangan acak atau memberikan sandi lewat *(passphrase)* sebagai *session key* untuk pesan saat ini.
c) Pengirim mengenkripsi *session key* tersebut dengan kunci *public* masing-masing penerima. Hasil enkripsi *session key* ini menjadi awal dari pesan yang dikirim.
d) Pengirim mengenkripsi pesan (yang biasanya sudah dikompresi) yang akan dikirim dengan menggunakan *session key.*
e) Penerima mendekripsi pesan dengan kunci privatnya.
f) Penerima mendekripsi pesan dengan *session key.* Jika pesan yang diterima merupakan hasil kompresan, maka pesan harus didekompresi.

3.7 Perintah Dasar GPG *(Gnu Privacy Guard)* Pada Linux Ubuntu

Tabel 3. 4 Perintah dasar GPG berbasis linux

PERINTAH	KETERANGAN
-s, --sign	Membuat tanda tangan
--clear-sign	Menghapus/membersihkan tanda tangan
-b, --detach-sign	Membuat tanda tangan yang objektif
-e, --encrypt	Enkripsi data
-c, --symmetric	Enkripsi data simetris
-d, --decrypt	Dekripsi data
--verify	Verifikasi tanda tangan
-k, --list-keys	Melihat daftar kunci
--list-signatures	Melihat daftar tanda tangan
--check-signatures	Memeriksa dan melihat tanda tangan
--fingerprint	Melihat daftar kunci (sidik jari)
-K, --list-secret-keys	Melihat daftar kunci pribadi
--generate-key	Menghasilkan pasangan kunci baru
--quick-generate-key	Menghasilkan pasangan kunci baru dengan cepat
--quick-add-uid	Menambahkan id pengguna baru dengan cepat
--quick-revoke-uid	Mencabut id pengguna dengan cepat
--quick-set-expire	Mengatur tanggal kedaluwarsa baru

--full-generate-key	Membuat pasangan kunci berfitur lengkap
--generate-revocation	Mencabut sertifikat
--delete-keys	Menghapus kunci publik
--delete-secret-keys	Menghapus kunci pribadi
--quick-sign-key	Menandatangani kunci dengan cepat
--quick-lsign-key	Menandatangani kunci lokal dengan cepat
--quick-revoke-sig	Mencabut tanda tangan kunci dengan cepat
--sign-key	Menandatangani kunci
--lsign-key	Menandatangani kunci lokal
--edit-key	Merubah kunci
--change-passphrase	Mengganti frasa sandi
--export	Mengekspor kunci
--send-keys	Mengekspor kunci ke server kunci
--receive-keys	Mengimpor kunci dari server kunci
--search-keys	Mencari kunci pada server kunci
--refresh-keys	Memperbarui semua kunci dari server kunci
--import	Mengimpor/gabungan kunci
--card-status	Mencetak status kartu
--edit-card	Ubah data pada kartu
--change-pin	Mengubah pin kartu
--update-trustdb	Memperbarui database
--print-md	Mencetak intisari pesan
--server	Menjalankan mode server
--tofu-policy VALUE	Mengatur kebijakan tofu untuk sebuah kunci
Options	
-a, --armor	Membuat keluaran dengan extensi ascii
-r, --recipient USER-ID	Mengenkripsi untuk user id penerima
-u, --local-user USER-ID	Menggunakan user id lokal untuk menandatangani dan mendekripsi
-z N	Mengatur tingkat kompresi ke N (0 menonaktifkan)
--textmode	Menggunakan mode teks kanonik
-o, --output FILE	Digunakan untuk mengeluarkan *output file*
-v, --verbose	Kata kerja
-n, --dry-run	Jangan membuat perubahan apa pun
-i, --interactive	Perintah interaktif sebelum menulis
--openpgp	Penggunaan kompabilitas pgp