# BAB III
# METODE PENELITIAN

## 3.1 Tempat dan Waktu Penelitian

Tempat penelitian merujuk pada sumber dimana peneliti gunakan untuk mengumpulkan data yang diperlukan selama melakukan penelitian. Sementara waktu penelitian merujuk pada jadwal pelaksanaan penelitian. *Detail* tentang tempat dan waktu penelitian akan dijelaskan pada sub-bab berikut ini.

### 3.1.1 Tempat Penelitian

Pada penelitian ini, tempat penelitian dan pengumpulan data menjadi tidak terikat, karena sifat dari penelitian ini yang menggunakan pendekatan rancang bangun sistem. Pendekatan ini melibatkan proses menerjemahkan hasil analisis menjadi suatu perangkat lunak, yang selanjutnya digunakan untuk menciptakan suatu sistem yang baru.

### 3.1.2 Waktu dan Jadwal Pelaksanaan Penelitian

Waktu dan jadwal pelaksanaan pada penelitian ini, lebih jelasnya dapat dilihat pada Tabel 3.1 berikut :

**Tabel 3.1** Jadwal Pelaksanaan Penelitian

| No. | Jenis Kegiatan | Jadwal Penelitian | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Maret 2024 | | April 2024 | | | | Agustus 2024 | | | | September 2024 | | | | Oktober 2024 | | | | November 2024 | | | | Desember 2024 | | | |
| | | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 |
| 1 | Studi Literatur | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 2 | Perencanaan | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 3 | Analisis Kebutuhan | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 4 | Perancangan Sistem | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 5 | Implementasi Sistem | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 6 | Pengujian Sistem | | | | | | | | | | | | | | | | | | | | | | | | | | |

## 3.2 Bahan dan Alat Penelitian

Bahan dan alat yang digunakan dalam penelitian ini meliputi sebuah unit laptop yang terdiri dari perangkat keras dan perangkat lunak. Detail tentang perangkat keras dan perangkat lunak yang digunakan pada penelitian ini akan dijelaskan pada sub-bab berikut ini.

### 3.2.1 Perangkat Keras

Perangkat keras merupakan komponen fisik alat elektronik yang membentuk bagian dari sistem komputer. Perangkat keras komputer mencakup berbagai macam komponen elektronik yang bekerja bersama untuk menjalankan program, memproses data, dan melakukan tugas lainnya. Berikut adalah spesifikasi kebutuhan perangkat keras pada penelitian ini :

**Tabel 3.2** Informasi Perangkat Keras Yang Digunakan

| No. | Nama Perangkat Keras | Spesifikasi |
|---|---|---|
| 1. | *Processor* | AMD Ryzen 3 3250U |
| 2. | ROM | Kingston SSD NVMe 512 GB |
| 3. | RAM | Kingston SODIMM 8 GB |
| 4. | *Graphic Card* | AMD Radeon(TM) Vega 3 |

### 3.2.2 Perangkat Lunak

Perangkat lunak merupakan bagian dari sistem komputer yang tidak berwujud secara fisik dan terdiri dari program atau instruksi yang memberikan arahan kepada perangkat keras untuk melakukan dan menyelesaikan tugas tertentu. Berikut adalah kebutuhan perangkat lunak pada penelitian ini :



**Tabel 3.3** Informasi Perangkat Lunak Yang Dibutuhkan

| No. | Nama Perangkat Lunak | Versi |
|---|---|---|
| 1. | *Operating System* | *Windows 10 Home 64-bit* |
| 2. | *Balsamiq Wireframes* | 4.7.4 |
| 3. | HTML | HTML5 |
| 4. | CSS | CSS3 |
| 5. | JavaScript | ECMAScript 2024 |

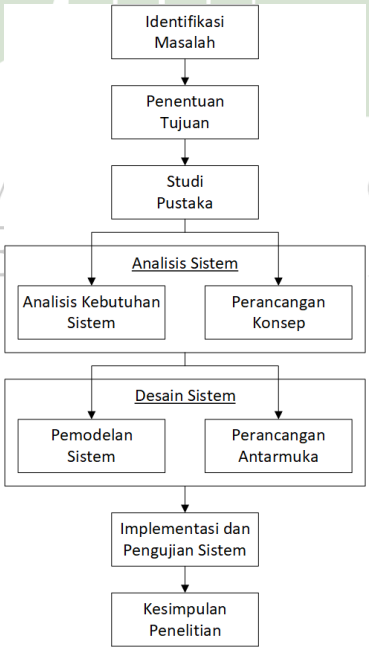
| No. | Nama Perangkat Lunak | Versi |
|---|---|---|
| 6. | *Visual Studio Code* | 1.95.3 |
| 7. | *Google Chrome* | 131.0.6778.86 |

## 3.3 Kerangka Kerja

Kerangka kerja merupakan serangkaian tahapan yang dilakukan dalam penelitian ini. Rincian kerangka kerja dapat dijelaskan pada sub-bab berikut ini.

### 3.3.1 Perencanaan

Tahapan perencanaan penelitian yang tepat diperlukan untuk mendukung proses penelitian tentang pengamanan pada *file source code* menggunakan algoritma kriptografi supaya sistematis dan terarah, diperlukan tahapan perencanaan penelitian yang tersusun dengan baik. Tahapan ini menjadi dasar dari penelitian dan dapat mempermudah dalam penulisan proposal penelitian. Tahapan perencanaan penelitian bertujuan untuk menguraikan masalah dan tujuan penelitian. Terdapat beberapa hal yang dilakukan selama proses perencanaan yakni identifikasi masalah, penentuan tujuan penelitian, studi pustaka, analisis sistem, desain sistem, implementasi dan pengujian sistem dan kesimpulan penelitian. Adapun tahapan perencanaan pada penelitian ini, terlihat pada Gambar 3.1 :




**Gambar 3.1** Tahapan Penelitian

Berikut ini merupakan penjelasan tentang tahapan penelitian, yang ditunjukkan pada Gambar 3.1 :

1. Identifikasi Masalah

   Pada tahap identifikasi masalah, yang merupakan langkah awal dan persiapan dalam sebuah penelitian, yang telah dilakukan identifikasi terlebih dahulu tentang apa saja yang menjadi permasalahan yang kemudian dikumpulkan dan dijadikan topik dari penelitian untuk dicari solusinya dan di jadikan tujuan dari penelitian ini. Dalam konteks penelitian ini, permasalahan yang diangkat terkait dengan mengamankan *file source code*. Pentingnya menjaga kerahasiaan dan integritas dari *file source code* tidak bisa diremehkan, mengingat risiko diakses oleh pihak yang tidak berwenang dan modifikasi yang tidak sah, baik saat disimpan dalam penyimpanan *hard drive* maupun saat akan dikirim. Kriptografi menjadi solusi yang efektif dalam mengamankan data atau informasi pada *file source code*. Dalam penelitian ini, megimplementasikan algoritma *Base64*, dan algoritma kriptografi simetris yaitu *Beaufort Cipher* dan *Vigenere Cipher*. Identifikasi masalah ini berisi pernyataan atau pertanyaan yang jelas terkait dengan penelitian.

2. Penentuan Tujuan

   Penentuan tujuan memperkuat urgensi dilakukannya penelitian, sehingga pemecahan masalah yang dirumuskan dapat tercapai setelah penelitian selesai dilakukan. Setelah masalah diidentifikasi, selanjutnya menetapkan tujuan penelitian yaitu untuk menyelesaikan masalah yang ada dengan membangun suatu aplikasi yang dapat mengamankan data atau informasi pada *file source code* menggunakan algoritma *Base64*, dan algoritma kriptografi simetris yaitu *Beaufort Cipher* dan *Vigenere Cipher*.



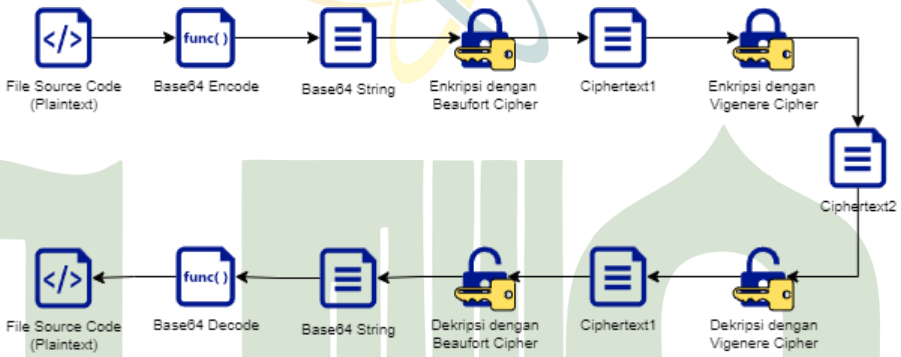
3. Studi Pustaka

   Studi pustaka adalah langkah yang penting dalam mendukung landasan teori penelitian, dengan memberikan informasi yang relevan terkait penelitian yang sedang dilakukan. Proses studi pustaka melibatkan pencarian informasi dari berbagai sumber, seperti menjelajahi di internet, membaca literatur seperti buku dan jurnal, menelusuri hasil penelitian terdahulu, serta sumber-sumber lain yang berkaitan dengan topik penelitian. Informasi yang dicari mencakup topik seperti

algoritma *Base64*, *Beaufort Cipher* dan *Vigenere Cipher*, dari penelitian sebelumnya yang relevan dengan penelitian ini. Hasil dari studi pustaka ini akan dijelaskan lebih lanjut dalam tinjauan pustaka.

4. Analisis Kebutuhan Sistem dan Perancangan Konsep

   Sebelum merancang sebuah sistem, tahapan yang harus dilakukan adalah menganalisis sistem yang akan dibuat. Analisis sistem mencakup kebutuhan fungsional dan non-fungsional. Setelah analisis sistem selesai, langkah berikutnya adalah merancang konsep pengamanan data atau informasi pada *file source code*. Konsep pengamanan yang dirancang, dengan mengimplementasikan algoritma *Base64*, *Beaufort Cipher* dan *Vigenere Cipher*. Rancangan konsep pengamanan tersebut dapat dilihat pada Gambar 3.2 :




**Gambar 3.2** Arsitektur Sistem

Arsitektur sistem pada Gambar 3.2, terdapat 13 tahapan dalam proses enkripsi dan dekripsi pada *file source code* menggunakan kriptografi dengan algoritma *Base64*, *Beaufort Cipher* dan *Vigenere Cipher*. Tahap pertama dalam proses enkripsi dimulai dengan menyiapkan suatu *file source code* (*plaintext*) yang akan dienkripsi. Berikutnya, mengubah teks di dalam *file source code* tersebut ke dalam format *Base64* melalui proses *Base64 encode*, yang kemudian menghasilkan *Base64 string*. Selanjutnya, menetapkan kunci yang akan digunakan untuk proses enkripsi dengan algoritma *Beaufort Cipher*, dan melakukan enkripsi tahap pertama yang menghasilkan *ciphertext1*. Kemudian *ciphertext1* dienkripsi lagi meng-gunakan kunci yang akan digunakan untuk proses enkripsi dengan algoritma *Vigenere Cipher*, dan melakukan enkripsi tahap kedua yang menghasilkan *ciphertext2*. *Ciphertext2* ini yang menjadi hasil dari *file source code* yang telah ter-enkripsi. Terakhir, dalam proses dekripsi tahap pertama adalah mendapatkan *ciphertext2*

yang kemudian di dekripsi dengan algoritma *Vigenere Cipher*, untuk mendapatkan *ciphertext1*. Selanjutnya *ciphertext1* di dekripsi lagi dengan algoritma *Beaufort Cipher*, untuk mendapatkan *Base64 string*. Kemudian *Base64 string* diubah menjadi teks biasa melalui proses *Base64 decode* untuk mendapatkan kembali *file source code* semula yang asli.

5. Desain Sistem

   Tahapan desain sistem dikerjakan setelah memperoleh pemahaman yang jelas tentang langkah-langkah yang didefinisikan dalam analisis sistem, dan kemudian meninjau untuk membangun sistem tersebut. Perancangan sistem mencakup pembuatan *flowchart* sistem dan desain tampilan antarmuka sistem yang akan diintegrasi dengan aplikasi di tahap implementasi sistem.

6. Impementasi dan Pengujian Sistem

   Tahapan implementasi dikerjakan dengan mengetikkan kode program yang membangun aplikasi tersebut yang merujuk pada desain sistem yang telah dibuat sebelumnya, menggunakan teknologi seperti HTML, CSS, dan JavaScript sebagai bahasa pemrograman untuk membangun aplikasi *web*. Setelah itu, dilakukan pengujian terhadap sistem yang telah dirancang untuk memastikan bahwa sistem berjalan sesuai dengan harapan. Metode pengujian yang digunakan adalah *black box testing*, yang merupakan metode untuk mempresentasikan fungsionalitas aplikasi saat digunakan, dengan memeriksa apakah *input* diterima dengan benar dan apakah *output* yang dihasilkan sesuai yang ditentukan.

7. Kesimpulan Penelitian

   Tahapan terakhir dari penelitian adalah kesimpulan dari hasil penelitian yang telah dilakukan, yang sejalan dengan tujuan akhir penelitian berdasarkan analisis hingga pengujian sistem yang telah dilakukan.



### 3.3.2 Teknik Pengumpulan Data

Metode pengumpulan data dibutuhkan untuk mendapatkan informasi yang mendukung pemecahan masalah yang dibahas. Dalam konteks ini, metode pengumpulan data yang digunakan meliputi :

1. Studi Pustaka

   Pengumpulan data dengan metode studi pustaka, di dapat melalui pencarian informasi dari berbagai sumber, seperti menjelajahi di internet, membaca literatur seperti buku dan jurnal, menelusuri hasil penelitian terdahulu, serta sumber-sumber lain yang berkaitan dengan topik penelitian ini.

2. Observasi

   Observasi dibuat untuk mengumpulkan informasi terkait format *file source code* bahasa pemrograman. Format *file source code* yang akan diamankan, dapat terdiri dari banyak bahasa pemrograman, dan yang menjadi intinya adalah isi dari *file source code* berbasis teks (*string*) dengan variasi ukuran *file* dan jumlah karakter. Hal ini tujuannya adalah untuk mengevaluasi kinerja dari algoritma *Base64*, *Beaufort Cipher* dan *Vigenere Cipher*.

### 3.3.3 Analisis Kebutuhan

Tahapan analisis kebutuhan dimaksudkan untuk mendapatkan spesifikasi tentang kebutuhan sistem. terdapat dua bagian, yaitu kebutuhan fungsional sistem dan kebutuhan non-fungsional sistem. Kebutuhan fungsional sistem menjelaskan fungsi-fungsi yang harus ada pada sebuah sistem untuk mencapai tujuannya. Sementara itu, kebutuhan non-fungsional sistem merujuk pada fitur-fitur lain seperti karakteristik, batasan sistem, dokumentasi, dan aspek lain yang diperlukan agar sistem dapat berjalan lancar.

1. Analisis Kebutuhan Fungsional

   Kebutuhan fungsional adalah kebutuhan terkait dengan proses yang akan dilakukan oleh sistem, serta mencakup informasi tentang fungsi-fungsi yang harus ada dan dihasilkan oleh sistem tersebut. Berikut adalah kebutuhan fungsional yang harus ada pada aplikasi pengamanan *file source code* yang mengimplementasikan algoritma *Base64*, *Beaufort Cipher* dan *Vigenere Cipher*, dapat dilihat pada Tabel 3.4 berikut :

**Tabel 3.4** Analisis Kebutuhan Fungsional

| No. | Kebutuhan Fungsional | Keterangan |
|---|---|---|
| 1. | Menerima *Input File Source Code* | Sistem dapat menerima berbagai format *file source code* yang diunggah dari perangkat pengguna, dan membaca isi dari *file* tersebut berupa teks (*string*). |
| 2. | Menerima *Input* Kunci | Sistem dapat menerima *input* kunci untuk algoritma *Beaufort Cipher* dan *Vigenere Cipher*. Kunci tersebut di *input* oleh pengguna dengan memasukkan sejumlah huruf atau karakter sesuai keinginan, dimana kunci tersebut hanya terdiri dari karakter *Base64 alphabet* tanpa spasi. Kunci yang digunakan antara algoritma *Beaufort Cipher* dan *Vigenere Cipher*, ditetapkan menggunakan kunci yang sama agar mempermudah manajemen kunci. |
| 3. | Mengenkripsi *File Source Code* | Sistem dapat mengenkripsi isi dari *file source code* yang berupa teks (*string*) dengan menggunakan algoritma *Base64*, *Beaufort Cipher* dan *Vigenere Cipher*. Langkah pertama yang dilakukan sistem adalah melakukan *Base64 encode* terhadap teks (*string*) dari isi *file source code*, yang akan menghasilkan *Base64 string*, kemudian *Base64 string* akan di enkripsi dengan algoritma *Beaufort Cipher*, dan akan menghasilkan *ciphertext1*, selanjutnya *ciphertext1* akan di enkripsi lagi dengan algoritma *Vigenere Cipher*, dan akan menghasilkan *ciphertext2* yang merupakan hasil akhir enkripsi atau *file source code* yang telah terenkripsi. |
| 4. | Mengunduh Hasil Enkripsi Dan Dekripsi *File Source Code* | Pada sistem terdapat fitur untuk mengunduh *file source code* yang telah terenkripsi dan terdekripsi, dimana hanya dengan mengklik tombol unduh, maka *file source code* yang telah terenkripsi atau terdekripsi akan mulai di unduh ke perangkat pengguna. |

| No. | Kebutuhan Fungsional | Keterangan |
|---|---|---|
| 5. | Mengunduh *Log* Kegiatan Enkripsi *File Source Code* | Pada sistem terdapat fitur untuk mengunduh *file log* kegiatan setelah melakukan enkripsi *file source code*, dimana ini dibuat agar dapat melihat rekaman kegiatan dari mengenkripsi *file source code*. |
| 6. | Mendekripsi *File Source Code* | Sistem dapat mendekripsi *file source code* yang terenkripsi (*ciphertext*) dengan menggunakan kunci yang digunakan pengguna pada tahap enkripsi sebelumnya, dengan cara mengunggah *file source code* yang terenkripsi yaitu *ciphertext2* yang kemudian di dekripsi dengan algoritma *Vigenere Cipher*, untuk mendapatkan *ciphertext1*. Selanjutnya *ciphertext1* di dekripsi lagi dengan algoritma *Beaufort Cipher*, untuk mendapatkan *Base64 string*. Kemudian *Base64 string* diubah menjadi teks biasa melalui proses *Base64 decode* untuk mendapatkan kembali *file source code* asli. |

2. Analisis Kebutuhan Non-Fugsional

   Kebutuhan non-fungsional merujuk pada persyaratan yang tidak secara langsung terkait dengan fungsionalitas utama suatu sistem, tetapi lebih kepada bagaimana sistem tersebut harus berperforma atau memenuhi aspek tertentu terkait dengan kualitasnya. Berikut adalah kebutuhan non-fungsional sistem dari aplikasi pengamanan *file source code* yang mengimplementasikan algoritma *Base64*, *Beaufort Cipher* dan *Vigenere Cipher*, dapat dilihat pada Tabel 3.5 berikut :



**Tabel 3.5** Analisis Kebutuhan Non-Fungsional

| No. | Kebutuhan Non-Fungsional | Keterangan |
|---|---|---|
| 1. | Terdapat Dokumentasi | Pada sistem dilengkapi menu *help* yang berisi panduan penggunaan aplikasi sehingga pengguna dapat menggunakannya sesuai petunjuk yang disediakan. |

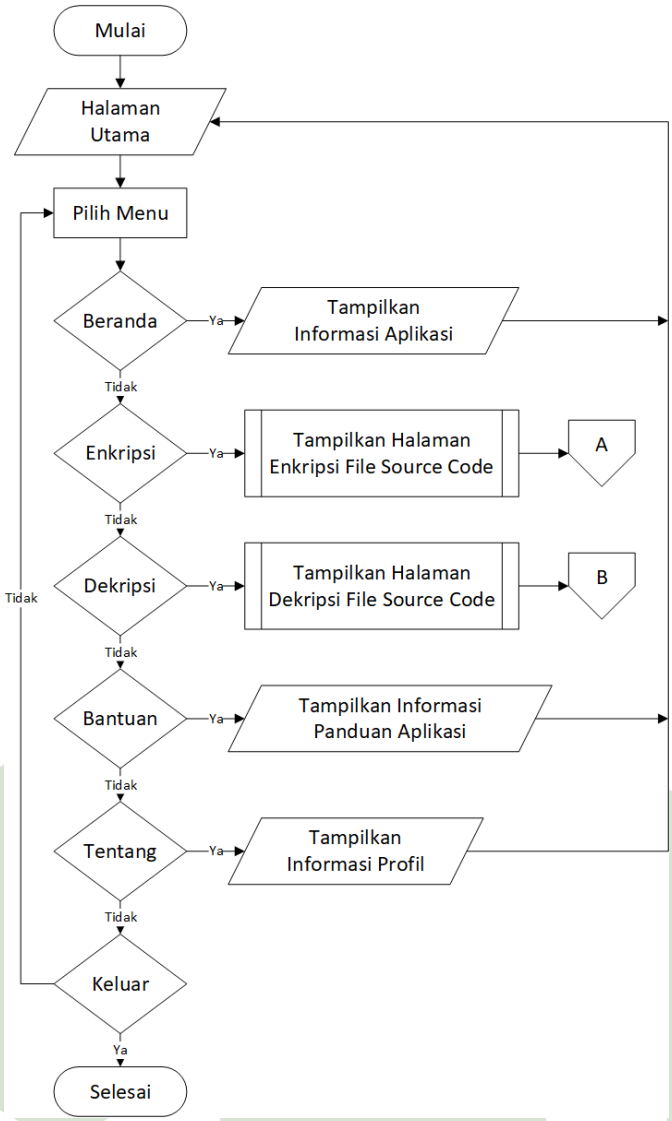
| No. | Kebutuhan Non-Fungsional | Keterangan |
|---|---|---|
| 2. | Ramah Pengguna (*User Friendly*) | Pada sistem menerapkan tampilan antarmuka yang mudah digunakan dan dipahami oleh pengguna, agar membuat pengalaman pengguna menjadi sederhana. |
| 3. | Penanganan *Error* (*Error-Handling*) | Pada sistem dilengkapi dengan penanganan *error*, dan akan menampilkan pesan *error* untuk setiap *input* dari pengguna yang tidak sesuai. |
| 4. | Kinerja | Pada sistem mampu melakukan proses enkripsi dan dekripsi pada *file source code* dengan komputasi yang cepat dan hanya membutuhkan waktu yang singkat. |

### 3.3.4 Perancangan

Perancangan sistem adalah implementasi teknis dari sistem yang direncanakan. Tujuannya adalah untuk menggambarkan semua kondisi dan komponen yang terlibat dalam sistem yang direncanakan. Dalam perancangan sistem ini, akan digunakan *flowchart* untuk menggambarkan alur kerja dari sistem dan *wireframe* untuk membuat rancangan antarmuka pengguna. Terdapat lima bagian yang butuh dirancang, yaitu terdiri dari halaman dari menu beranda, enkripsi, dekripsi, bantuan, dan tentang.

Perancangan *flowchart* sistem adalah representasi grafis yang menunjukkan alur kerja keseluruhan dari suatu sistem. *Flowchart* menjelaskan urutan langkah-langkah dari prosedur yang ada di dalam sistem. Dengan menggunakan *flowchart*, sistem dapat dipahami secara visual apa yang dilakukan oleh sistem dari aplikasi pengamanan *file source code* yang mengimplementasikan algoritma *Base64*, *Beaufort Cipher* dan *Vigenere Cipher*.

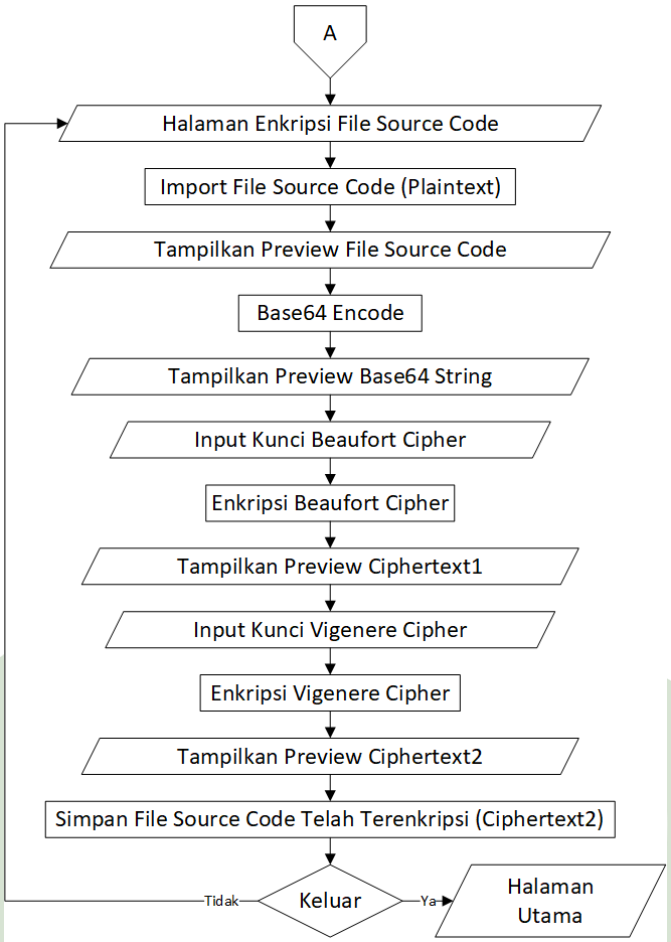
Adapun *flowchart* sistem dari aplikasi pengamanan *file source code* dapat dijelaskan seperti yang terlihat pada Gambar 3.3 :




**Gambar 3.3** *Flowchart* Sistem Dari Aplikasi Pengamanan *File Source Code*

Pada Gambar 3.3 menampilkan *flowchart* sistem yang dimulai dengan menampilkan halaman utama dari aplikasi. Pada halaman utama, terdapat beberapa menu yang dapat diakses oleh pengguna. Jika pengguna memilih menu "beranda", sistem akan menampilkan informasi tentang aplikasi. Jika memilih menu "enkripsi", sistem akan menampilkan halaman untuk melakukan proses enkripsi *file source code*. Jika memilih menu "dekripsi", sistem akan menampilkan halaman untuk melakukan proses dekripsi *file source code*. Jika memilih menu "bantuan", sistem akan menampilkan informasi panduan tentang cara menggunakan aplikasi. Jika memilih menu "tentang", sistem akan menampilkan informasi tentang profil pengembang aplikasi. Jika pengguna memilih menu "keluar", aplikasi akan ditutup.

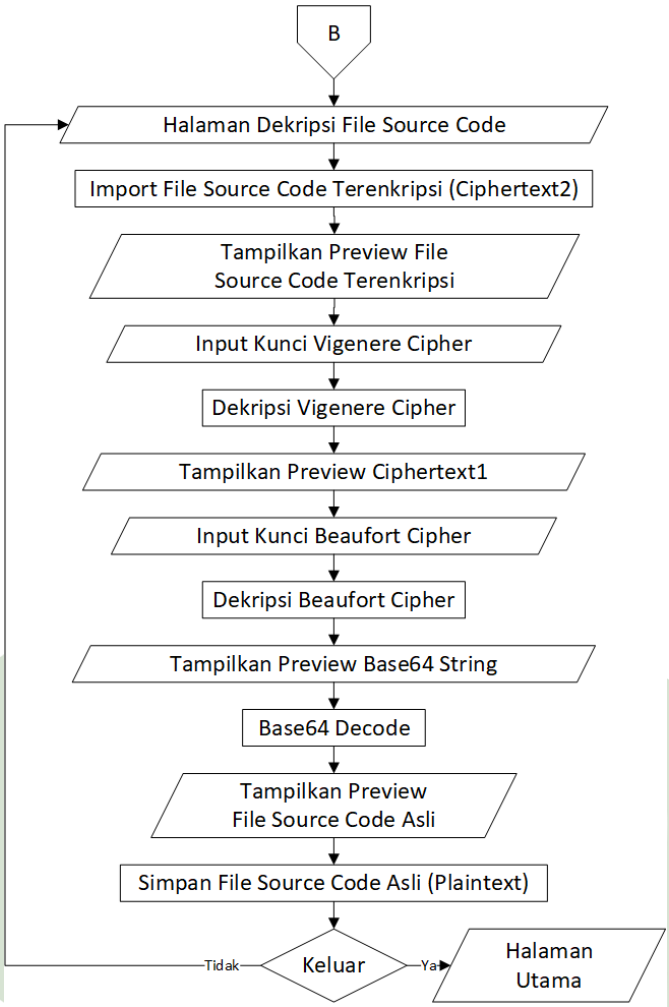
Adapun *flowchart* sistem dari halaman enkripsi *file source code*, dapat dijelaskan seperti yang terlihat pada Gambar 3.4 :




**Gambar 3.4** *Flowchart* Sistem Dari Halaman Enkripsi *File Source Code*

Pada Gambar 3.4 menggambarkan *flowchart* yang dimulai dengan menampilkan antarmuka halaman enkripsi *file source code*. Pada halaman ini, pengguna mengklik tombol "*input file*" untuk memasukkan *file source code* yang akan di enkripsi, dan isi dari *file source code* dapat dilihat pada *preview textarea*. Selanjutnya melakukan *Base64 encode* terhadap teks (*string*) dari isi *file source code*, dan akan menghasilkan *Base64 string*. Berikutnya, *Base64 string* akan di enkripsi dengan memasukkan kunci untuk algoritma *Beaufort Cipher*, dan akan menghasilkan *ciphertext1*, selanjutnya *ciphertext1* akan di enkripsi lagi dengan memasukkan kunci untuk algoritma *Vigenere Cipher*, dan akan menghasilkan *ciphertext2* yang merupakan hasil akhir enkripsi. Setelah itu simpan *file source code* yang telah terenkripsi untuk digunakan pada proses dekripsi.

Adapun *flowchart* sistem dari halaman enkripsi *file source code*, dapat dijelaskan seperti yang terlihat pada Gambar 3.5 :




**Gambar 3.5** *Flowchart* Sistem Dari Halaman Dekripsi *File Source Code*

Pada Gambar 3.5 menggambarkan *flowchart* yang dimulai dengan menampilkan antarmuka halaman dekripsi *file source code*. Pada halaman ini, pengguna mengklik tombol "*input file*" untuk memasukkan *file source code* yang akan di dekripsi, dan isi dari *file source code* dapat dilihat pada *preview textarea*. Selanjutnya melakukan dekripsi dengan memasukkan kunci untuk algoritma *Vigenere Cipher*, untuk mendapatkan *ciphertext1*. Selanjutnya *ciphertext1* di dekripsi lagi dengan memasukkan kunci untuk algoritma *Beaufort Cipher*, untuk mendapatkan *Base64 string*. Kemudian *Base64 string* diubah menjadi teks biasa melalui proses *Base64 dencode* untuk mendapatkan kembali *file source code* semula yang asli dengan mengklik tombol "unduh *file*".