**BAB III**

**METODE PENELITIAN**

* 1. **Objek Penelitian**

Penulis melakukan penelitian terhadap perancangan aplikasi kriptografi enkripsi dokumen untuk meningkatkan keamanan data dokumen Word dengan mengkombinasi metode *vernam cipher* dan *hill cipher.* Aplikasi kriptografi ini secara umum memiliki dua fungsi utama yaitu mengenkripsi file format docx dan mendekripsikannya kembali.

* 1. **Metode Pengumpulan Data**

Metode yang penulis gunakan dalam pengumpulan data adalah studi pustaka dan *Research and Site Visits* (Penelitian dan mengunjungi situs). Dimana Studi pustaka merupakan studi pendahuluan yang bermanfaat untuk menegetahui penelitian terdahulu terhadap masalah kriptografi dengan metode *vernam cipher* dan *hill cipher*. Pada Pustaka dimana peneliti mengambil buku-buku maupun jurnal yang terkait dengan Penelitian. Seperti halnya tentang Perancangan aplikasi kriptografi, jurnal yang terkait dengan dengan analisis kombinasi metode *vernam cipher* dan *hill cipher*, dan juga jurnal yang berkaitan dengan *prototype.*

* 1. **Alat dan Bahan Perlengkapan**

**3.2.1 Kebutuhan Perangkat Keras**

Sebelum memasuki tahapan pembuatan aplikasi enkripsi dokumen berbasis web, ada beberapa kebutuhan perangkat keras yang digunakan dalam pengembangan aplikasi *enkripsi* dokumen dengan mengkombinasi metode *vernam* *cipher* dan *hill* *cipher*, yaitu:

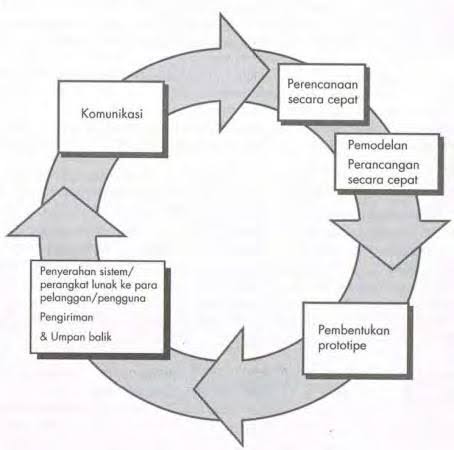
1. Laptop/Notebook
2. Prosesor: Intel Celeron N2840
3. RAM: 2 GB DDR3
4. LCD: 14TFT LCD
5. Penyimpanan 500GB
6. *System Manufacturer* : LENOVO

**3.2.2 Kebutuhan Perangkat Lunak**

Perangkat lunak yang digunakan dalam penerapan enkripsi dan dekripsi dokumen Word menggunakan kombinasi metode vernam cipher dan hill cipher adalah sebagai berikut:

1. Sistem Operasi *Windows 10 64-bit*
2. *XAMPP*
3. *Google Chrome*
4. *Sublime Text 3*
5. *Microsoft Office Word 2019*
6. *Microsoft Office Visio 2019*
   1. **Pengembangan Perangkat Lunak *Prototype***

Metode pengembangan yang digunakan dalam perancangan aplikasi kriptografi adalah metode *Prototype.* Metode *prototype* adalah metode dalam pengembangan rekayasa *software* yang bertahap dan berulang, serta mementingkan sisi *user* sistem. Metode Pengembangan Aplikasi yang digunakan untuk membangun aplikasi kriptografi enkripsi dokumen yaitu dengan menggunakan metode prototype yang terdiri dari 5 tahapan pengembangan perangkat lunak yaitu Tahap Komunikasi, Tahap Perencanaan secara cepat, Tahap Pemodelan Perancangan secara cepat, Tahap Pembentukan Prototype, Tahap Pembuatan Sistem Serta Penyerahan & Umpan Balik. Tahapan pengembangan perangkat lunak model prototype pada penelitian ini dapat dilihat pada gambar 3.1.



Gambar 3.1 Tahapan Pengembangan Perangkat Lunak Model *Prototype*

Berikut ini penjelasan tahapan-tahapan pengembangan perangkat lunak model *prototype*:

1. Tahap Komunikasi

Pada tahap ini, peneliti mengidentifikasi spesifikasi kebutuhan yang dibutuhkan dengan cara menggunakan laptop/notebook secara langsung untuk memperoleh data-data yang dibutuhkan serta mengumpulkan data-data dari studi pustaka dan *Research and Site Visits* (Penelitian dan mengunjungi situs). Pada Pustaka dimana peneliti mengambil jurnal-jurnal yang terkait dengan Metode *Vernam Cipher dan Hill Cipher* pada enkripsi file, seperti halnya tentang Kriptografi *Vernam Cipher* Untuk Mencegah Pencurian Data Pada Semua Ekstensi File, jurnal yang terkait dengan dengan Pengamanan dokumen *Office*. Kemudian hasilnya akan dianalisa untuk menentukan kebutuhan sistem.

1. Tahap Perencanaan secara cepat

Pada tahap ini peneliti akan mendefinisikan batas waktu dengan membuat jadwal penelitian untuk perancangan aplikasi kriptografi enkripsi dokumen, Kemudian peneliti akan mendefinisikan informasi yang berhubungan dengan aplikasi ini dan menentukan tools apa saja yang akan digunakan. Tool yang akan digunakan dalam pembuatan aplikasi ini berupa *XAMPP, Notepad++*, *microsoft office word 2019*, *microsoft office visio 2019, Google Chrome.*

1. Tahap Pemodelan Perancangan Secara Cepat

Pada tahap ini peneliti akan menganalisa kebutuhan-kebutuhan perangkat lunak yang akan di butuhkan nantinya dan peneliti akan mendesain cara kerja dari sistem perancangan aplikasi kriptografi enkripsi dokumen yang akan di buat. Seperti perancangan *prototype* dari Aplikasi Kriptografi Enkripsi dan Dekripsi Dokumen Word yang akan dibuat terhadap permasalahan yang ada, dengan menggunakan

perangkat permodelan *Unified Modeling Language* (UML).

1. Tahap Pembentukan *Prototype*

Pada tahap ini peneliti akan melakukan penulisan kode kedalam program dengan menggunakan aplikasi *Notepad++* sesuai dengan tahap perancangan yang telah dibuat dan tools yang telah ditentukan sebelumnya. Kemudian akan dilakukan pengujian sistem dengan *Black box* *testing* untuk mengetahui apakah sistem aplikasi kriptografi enkripsi dokumen *Word* tersebut masih ditemukan kesalahan sehingga harus diperbaiki atau tidak. Tahap pengujian ini juga bertujuan untuk menunjukan apakah masukan data dan keluaran data telah berjalan sebagaimana yang diharapkan atau tidak.

1. Tahap Pembuatan Sistem serta Penyerahan & Umpan Balik

Sistem yang sudah lulus tahapan pengujian dibuat prototype agar dapat diakses dan digunakan oleh pengguna atau pengembang sistem.

* 1. **Perancanagan Aplikasi**

Perancangan aplikasi disini adalah bagaimana peneliti menggambararkan sebuah perangkat lunak kriptografi untuk enkripsi dan dekripsi dokumen format docx menggunakan kombinasi metode *vernam cipher* dan *hill cipher.* Pada tahap perancangan peneliti melakukan pemodelan sistem kriptografi enkripsi dan dekripsi file dengan menggunakan *Unified Modeling Language* (UML).

* + 1. ***Use Case Diagram* Utama**

*Use case* diagram utama menggambarkan tentang halaman Enkripsi dan Dekripsi dari aplikasi enkripsi file dokumen. Fitur yang terdapat dalam menu Enkripsi yaitu upload file, Input kunci enkripsi, tombol untuk enkripsi file dokumen, Tampilan hasil enkripsi dan tombol untuk menyimpan file enkripsi. Sedangkan fitur pada menu dekripsi tidak jauh berbeda dengan menu enkripsi. Adapun tampilan *use case* diagram utama dapat dilihat pada gambar 3.2 dan gambar 3.3.



Gambar 3.2 *Use Case* Menu Enkripsi



Gambar 3.3 *Use Case* Menu Dekripsi

* + 1. ***Activity Diagram* Utama**

Proses yang akan dijelaskan pada gambar dibawah ini yaitu saat pengguna mengakses aplikasi enkripsi dokumen. Ketika pengguna menjalankan aplikasi tersebut untuk pertama kalinya, maka pengguna akan diarahkan ke menu beranda. Proses selanjutnya tergantung pada kebutuhan pengguna. Apabila pengguna ingin mengenkripsi sebuah dokumen maka pengguna akan masuk ke menu enkripsi untuk melakukan proses enkripsi dokumen.

Sama halnya dengan menu dekripsi, jika pengguna ingin mendekripsi dokumen yang telah dienkripsi tersebut, maka langkah yang harus dilakukan oleh pengguna yaitu masuk ke menu dekripsi, kemudian meng-upload file yang telah dienkripsi oleh aplikasi tersebut, lalu pengguna memasukan kunci dekripsi (kunci yang sama ketika pengguna melakukan enkripsi file), setelah itu melakukan proses dekripsi pada file terenkripsi dengan cara menekan tombol *decrypt.*

Rangkaian aktivitas antara pengguna dan aplikasi kriptodocx (aplikasi enkripsi dan

dekripsi file docx) dapat dilihat pada gambar 3.4.



Gambar 3.4 *Activity Diagram* Utama

* 1. **Langkah-Langkah Enkripsi Dan Dekripsi Pada *Vernam Cipher***

Metode *Vernam Cipher* merupakan sistem kerahasiaan yang sempurna di mana metode ini adalah *stream cipher* simetris di mana *plaintext* dikombinasikan dengan *key* *stream* (*pseudorandom*) yang sama panjang untuk menghasilkan *ciphertext* yang mengfungsikan *boolean* *ekslusif* (Puspita & Wayahdi, 2015). Proses Enkripsi dan Dekripsi pada *Vernam Cipher* dilakukan menggukan rumus seperti berikut ini:

Rumus Enkripsi Pesan:

Rumus Dekripsi Pesan:

Dimana:

Pi = *Plaintext* (Pesan asli)

Ki = *Key* (Kunci)

Pi = *Ciphertext* (Pesan Rahasia)

= Operator Logika *XOR*

Contoh kasus dalam mengenkripsi dan dekripsi pada algoritma *Vernam Cipher* yaitu misalkan ada sebuah pesan yang berisi plainteks BILVAN, yang akan diproses dengan kunci XDXYVC. Sebelum melakukan enkripsi, Kita perlu menyepakati acuan huruf yang akan kita gunakan dalam proses enkripsi pesan, berikut ini tabel alfabetik 26 huruf yang dikonversi ke bilangan biner, dapat dilihat pada tabel 3.1.

Tabel 3.1 Alfabetik 26 huruf yang dikonversi ke biner

| **Karakter** | **Biner** | **Karakter** | **Biner** |
| --- | --- | --- | --- |
| A | 01000001 | N | 01001110 |
| B | 01000010 | O | 01001111 |
| C | 01000011 | P | 01010000 |
| D | 01000100 | Q | 01010001 |
| E | 01000101 | R | 01010010 |
| F | 01000110 | S | 01010011 |
| G | 01000111 | T | 01010100 |
| H | 01001000 | U | 01010101 |
| I | 01001001 | V | 01010110 |
| J | 01001010 | W | 01010111 |
| K | 01001011 | X | 01011000 |
| L | 01001100 | Y | 01011001 |
| M | 01001101 | Z | 01011010 |

Dari table 3.1 diatas, maka hal pertama yang harus dilakukan yaitu mengkonversikan plainteks dan kunci kedalam bilangan biner dan dilanjutkan dengan proses enkripsi dan dekripsi yang dapat dilihat pada tabel 3.2 dan tabel 3.3.

Tabel 3.2 Konversi plainteks ke bilangan biner

|  |  |
| --- | --- |
| **Plainteks** | **Bilangan Biner** |
| B | 01000010 |
| I | 01001001 |
| L | 01001100 |
| V | 01010110 |
| A | 01000001 |
| N | 01001110 |

Tabel 3.3 Konversi kunci ke bilangan biner

|  |  |
| --- | --- |
| **Kunci** | **Bilangan Biner** |
| X | 01011000 |
| D | 01000100 |
| N | 01001110 |
| Y | 01011001 |
| V | 01010110 |
| C | 01000011 |

Setelah itu proses enkripsi akan dilakukan dengan cara melakukan operasi *bitwise* XOR antara Plainteks dan kunci. Proses enkripsi dapat dilihat pada tabel 3.4.

Tabel 3.4 Proses enkripsi pada *vernam cipher*

|  |  |  |  |
| --- | --- | --- | --- |
| Plainteks | B | I | L |
| 01000010 | 01001001 | 01001100 |
| Kunci | 01011000 | 01000100 | 01001110 |
| Cipherteks | 00011010 | 00001101 | 00000010 |
| SUB | CR | STX |
| Plainteks | V | A | N |
| 01010110 | 01000001 | 01001110 |
| Kunci | 01011001 | 01010110 | 01000011 |
| Cipherteks | 00001111 | 00010111 | 00001101 |
| SI | ETB | CR |

Dari hasil enkripsi yang dapat dilihat pada tabel 3.4 diatas, cipherteks yang diperoleh dari plainteks BILVAN yaitu SUBCRSTXSIETBCR. Cipherteks tersebut bukanlah sebuah huruf melainkan sebuah karakter tidak terlihat yang dapat disimak pada tabel ASCII di bab sebelumnya. Untuk melakukan dekripsi pada cipherteks SUBCRSTXSIETBCR, maka langkah yang harus dilakukan yaitu dengan cara melakukan operasi *bitwise* XOR antara cipherteks dan kunci yang dapat dilihat pada tabel 3.5.

Tabel 3.5 Proses dekripsi pada *vernam cipher*

|  |  |  |  |
| --- | --- | --- | --- |
| Cipherteks | SUB | CR | STX |
| 00011010 | 00001101 | 00000010 |
| Kunci | 01011000 | 01000100 | 01001110 |
| Plainteks | 01000010 | 01001001 | 01001100 |
| B | I | L |
| Cipherteks | SI | ETB | CR |
| 00001111 | 00010111 | 00001101 |
| Kunci | 01011001 | 01010110 | 01000011 |
| Plainteks | 01010110 | 01000001 | 01001110 |
| V | A | N |

Dari hasil proses dekripsi cipherteks pada tabel 3.5 diatas, diperoleh plainteks asli yaitu BILVAN. Ini berarti proses enkripsi dan dekrisi algoritma *vernam cipher* yang telah dilakukan berhasil.

* 1. **Langkah-Langkah Enkripsi Dan Dekripsi Pada *Hill Cipher***

Proses enkripsi dan dekripsi pada algoritma *Hill Cipher* dilakukan per blok plainteks. Ukuran blok tersebut sama dengan blok matriks kunci. Sebelum membagi teks menjadi deretan blok-blok, plainteks terlebih dahulu dikonversi menjadi angka. Berikut ini akan diperlihatkan tabel huruf yang dikonversi kedalam bentuk angka pada tabel 3.6.

Tabel 3.6 Konversi huruf kedalam bentuk angka

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

Secara matematis, proses enkripsi pada *hill cipher* dapat dirumuskan sebagai berikut:

*C = K . P*

Keterangan:

*C = Ciphertext*

*K = Key*

*P = Plaintext*

Contoh kasus misalnya ingin menyembunyikan sebuah pesan berisi nama BILVANSIUS dengan kunci matriks ukuran 2 x 2, K = , kemudian bagi plainteks sesuai dengan ukuran matriks. Setelah menentukan kunci, maka langkah selanjutnya yaitu mengkonversi pesan tersebut ke bilangan desimal yang dapat dilihat pada tabel 3.7.

Tabel 3.7 Plainteks Dikonversi Ke Bilangan Desimal

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Blok I** | **Blok II** | **Blok III** | **Blok IV** | **Blok V** |
| B I | L V | A N | S I | U S |
| 1 8 | 11 21 | 0 13 | 18 8 | 20 18 |

Blok I

= =

Setelah itu hasil di mod 26 seperti berikut:

28 mod 26 = 2

27 mod 26 = 1

Hasilnya 2 dan 1 atau C dan B.

Blok II

= =

Setelah itu hasil di mod 26 seperti berikut:

107 mod 26 = 3

96 mod 26 = 18

Hasilnya 3 dan 18 atau D dan S.

Blok III

= = =

Setelah itu hasil di mod 26 seperti berikut:

39 mod 26 = 13

39 mod 26 = 13

Hasilnya 13 dan 13 atau N dan N.

Blok IV

= = =

Setelah itu hasil di mod 26 seperti berikut:

96 mod 26 = 18

78 mod 26 = 0

Hasilnya 18 dan 0 atau S dan A.

Blok V

= = =

Setelah itu hasil di mod 26 seperti berikut:

134 mod 26 = 4

114 mod 26 = 10

Hasilnya 4 dan 10 atau E dan K.

Maka hasil *ciphertext* dari *plaintext* BILVANSIUS adalah CBDSNNSAEK.

Untuk melakukan proses dekripsi yaitu dengan melakukan perkalian pada invers matriks kunci dengan blok matriks *ciphertext*, maka proses dekripsi sebagai berikut:

*K =*  Det(*K*) = (4\*3) – (3\*3) = 3

Invers *modulo* = 3-1 mod 26 = (3\*b) mod 26 = 1

*K* = *n* =

*K* = 0 = = = = 0,3

*K* = 1 = = = = 9

Jika sudah mendapat bilangan bulat pada perhitungan di atas, maka selanjutnya nilai invers modulo determinan tersebut digunakan untuk mencari invers matriks.

*K =*  maka *K-1 =*

Sehingga: *K-1 = 9 = mod* 26

Setiap bilangan yang bernilai negatif dapat dikerjakan dengan rumus -n mod x = x-(n mod x), sehingga nilai negatif pada matriks K-1 dapat dihitung seperti berikut:

-n mod x = x-(n mod x)

= 26 – (27 mod 26) = 25

*K-1 = mod* 26 =

Setelah matriks *K-1*  telah diketahui, selanjutnya mengalikan matriks *K-1* dengan ciphertext.

Blok I

= =

Setelah itu hasil di mod 26 seperti berikut:

27 mod 26 = 1

60 mod 26 = 8

Hasilnya 1 dan 8 atau B dan I.

Blok II

= =

Setelah itu hasil di mod 26 seperti berikut:

453 mod 26 = 11

255 mod 26 = 21

Hasilnya 11 dan 21 atau L dan V.

Blok III

= = =

Setelah itu hasil di mod 26 seperti berikut:

338 mod 26 = 0

455 mod 26 = 13

Hasilnya 0 dan 13 atau A dan N.

Blok IV

= = =

Setelah itu hasil di mod 26 seperti berikut:

18 mod 26 = 18

450 mod 26 = 8

Hasilnya 18 dan 8 atau S dan I.

Blok V

= = =

Setelah itu hasil di mod 26 seperti berikut:

254 mod 26 = 20

200 mod 26 = 18

Hasilnya 20 dan 18 atau U dan S.

Dari hasil proses dekripsi algoritma *hill cipher* diatas, maka *plaintext* dari *ciphertext* CBDSNNSAEK adalah BILVANSIUS.

* 1. **Algoritma Proses Kriptografi Kombinasi Metode *Vernam Cipher* Dan *Hill Cipher***

Ilmu matematika dan ilmu komputer menjelaskan bahwa algoritma merupakan urutan atau langkah-langkah untuk penghitungan atau untuk menyelesaikan suatu masalah yang ditulis secara berurutan. Sehingga, algoritma pemrograman adalah urutan atau langkah-langkah untuk menyelesaikan masalah pemrograman komputer.

Dalam pemrograman, hal yang penting untuk dipahami adalah logika kita dalam berpikir bagaimana cara untuk memecahkan masalah pemrograman yang akan dibuat. Sebagai contoh, banyak permasalahan matematika yang mudah jika diselesaikan secara tertulis, tetapi cukup sulit jika kita terjemahkan ke dalam pemrograman. Dalam hal ini, algoritma dan logika pemrograman akan sangat penting dalam pemecahan masalah.

Algoritma proses kriptografi kombinasi metode *Vernam Cipher* dan *Hill Cipher* merupakan langkah-langkah dari proses kriptografi (Enkripsi dan Dekripsi) dari gabungan kedua metode yang telah disebutkan. Dibawah ini akan ditunjuka algoritma dari proses kriptografi gabungan kedua metode tersebut:

1. Algoritma Enkripsi Kombinasi Metode *Vernam Cipher* dan *Hill Cipher*

|  |
| --- |
| 1. Mulai 2. Masukan Plaintext dan Kunci Enkripsi 3. Hitung jumlah karakter plaintext, jika jumlah Plaintext = genap, maka lanjut ke langkah 5, jika tidak, maka lanjut ke langkah 4 4. Tambahkan karakter bantuan “X” pada plaintext 5. Konversi Plaintext ke bilagan biner 6. Hitung Ciphertext1 = Plaintext XOR Kunci Enkripsi 7. Hasil Ciphertext1 dikonversi ke bilangan desimal 8. Bagi Ciphertext1 kedalam beberapa Blok matrix 9. Buat Kunci Enkripsi menjadi kunci matrix 10. Lakukan perkalian matrix disetiap blok matrix, Ciphertext2 = kunci matrix \* blok matrix 11. Hasil Ciphertext2 dikonversi ke karakter 12. Tampilkan hasil Ciphertext2 13. Selesai |

1. Algoritma Dekripsi Kombinasi Metode *Vernam* *Cipher* dan *Hill Cipher*

|  |
| --- |
| 1. Mulai 2. Masukan Ciphertext2 dan Kunci Dekripsi 3. Konversi Ciphertext2 ke bilangan desimal 4. Bagi Ciphertext2 kedalam beberapa blok matrix 5. Buat kunci dekripsi menjadi kunci matrix 6. Cari invers matrix kunci 7. Lakukan perkalian invers matrix kunci dengan blok matrix, Ciphertext1 = Invers matrix kunci \* blok matrix 8. Hasil Ciphertext1 dikonversi ke bilangan biner 9. Konversi Kunci Dekripsi ke bilangan biner 10. Hitung Plaintext = Ciphertext1 XOR Kunci Dekripsi 11. Hasil Plaintext dikonversi ke karakter 12. Tampilkan hasil plaintext 13. Selesai |

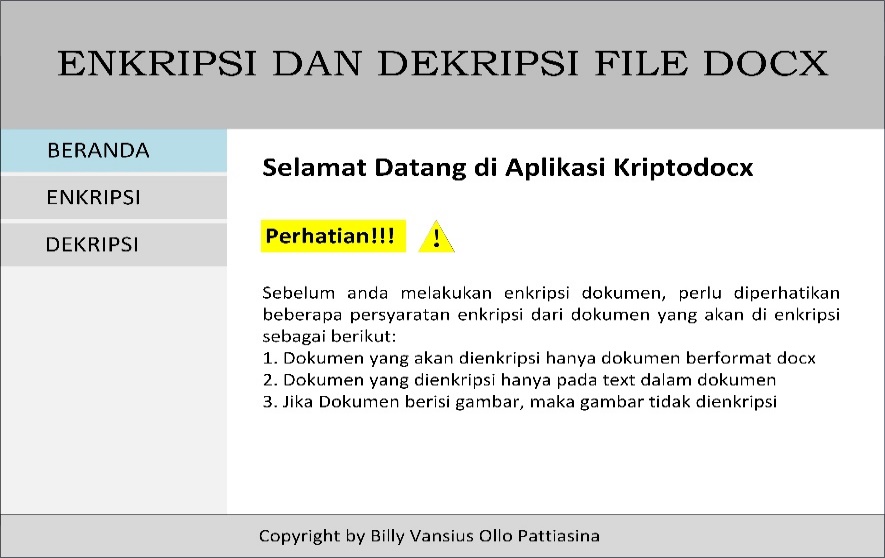
* 1. **Perancangan *Interface* (Antar Muka)**

Perancangan *interface* dilakukan untuk menggambarkan, merencanakan, dan membuat sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh dan berfungsi. Perancangan ini merupakan hasil transformasi dari analisa ke dalam perancangan yang nantinya akan akan diimplementasikan. Adapun rancangan tampilan aplikasi enkripsi dan dekripsi dokumen word dapat dilihat dibawah ini.

1. Halaman Beranda

Pada halaman ini hanya menampilkan pesan “Selamat Datang” dan beberapa persyaratan enkripsi dari dokumen yang akan dienkripsi. Halaman ini bertujuan agar sebelum pengguna melakukan enkripsi dokumen, pengguna perlu mengetahui beberapa persyaratan dari dokumen yang ingin di enkripsi. Adapun tampilan beranda yang dapat

dilihat pada gambar 3.5.



Gambar 3.5 Tampilan Beranda

1. Halaman Enkripsi

Halaman Enkripsi berfungsi untuk melakukan enkripsi dokumen yang di-*upload* oleh pengguna. Desain tampilan halaman enkripsi dapat dilihat pada gambar 3.6.



Gambar 3.6 Halaman Enkripsi

1. Halaman Dekripsi

Halaman dekripsi berfungsi untuk melakukan dekripsi dokumen yang terenkripsi dari aplikasi ini. Cara kerjanya sama seperti pada halaman enkripsi yaitu pengguna meng*-upload* dokumen yang telah dienkripsi sebelumnya, lalu memasukan kunci dan mendekripsinya. Berikut ini tampilan halaman dekripsi yang dapat dilihat pada gambar 3.7.



Gambar 3.7 Halaman Dekripsi

* 1. **Metode Pengujian Sistem**

Pada penelitian ini, metode pengujian yang akan dipakai dalam pengembangan aplikasi ini adalah *black box testing*. pengujian ini bertujuan untuk mengetahui apakah kebutuhan fungsional dari sistem sudah terpenuhi atau belum. Adapun rencana kasus uji pengujian fungsional *user* *interface* ini bias dilahat pada tabel 3.8.

Tabel 3.8 Pengujian fungsional *User Interface*

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Kasus Uji** | | **Hasil yang diharapkan** |
| Halaman Beranda | | | |
| 1 | Hyperlik navbar vertikal | | Dapat mengakses setiap halaman navbar vertikal (Beranda, Enkripsi, Dekripsi) |
| Halaman Enkripsi | | | |
| 1 | | Tombol *Upload* File | Dapat meng-*upload* file ke halaman web |
| 2 | | Textbox Kunci | Dapat meng-*input* angka ke dalam *textbox* |
| 3 | | Tombol Encrypt | Dapat mengenkripsi file yang di-*upload* |
| 4 | | Text area Chipertext | Dapat menampilkan hasil enkripsi dari isi file |
| 5 | | Tombol Save to docx | Dapat menyimpan hasil enkripsi ke dalam dokumen format docx |

|  |  |  |
| --- | --- | --- |
| Halaman Dekripsi | | |
| 1 | Tombol *Upload* File | Dapat meng-*upload* file ke halaman web |
| 2 | Textbox Kunci | Dapat meng-*input* angka ke dalam *textbox* |
| 3 | Tombol Decrypt | Dapat mendekripsi file terenkripsi yang telah di-*upload* |
| 4 | Text area Plaintext | Dapat menampilkan hasil dekripsi dari isi file |
| 5 | Tombol Save to docx | Dapat menyimpan hasil dekripsi ke dalam dokumen format docx |

* 1. **Jadwal Penelitian**

Waktu pelaksanaan penelitian diperkirakan dimulai pada awal bulan September sampai akhir bulan November 2019, lokasi penelitian di Kampus III Universitas Khairun. Adapun tabel jadwal penelitian yaitu dapat dilihat pada tabel 3.9.

Tabel 3.9 Jadwal Penelitian

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No | Kegiatan | Bulan ke-1 | | | | Bulan Ke-2 | | | | Bulan ke-3 | | | |
| 1. | Analisis Kebutuhan |  |  |  |  |  |  |  |  |  |  |  |  |
| 2. | Kebutuhan *Input/Output* |  |  |  |  |  |  |  |  |  |  |  |  |
| 3. | Perancangan Sistem |  |  |  |  |  |  |  |  |  |  |  |  |
| 4. | Pembuatan Sistem |  |  |  |  |  |  |  |  |  |  |  |  |
| 5. | Pengujian Sistem |  |  |  |  |  |  |  |  |  |  |  |  |
| 6. | Pembuatan Laporan |  |  |  |  |  |  |  |  |  |  |  |  |