BAB III

METODOLOGI DAN PERANCANGAN APLIKASI

3.1 Metodologi Penelitian

Metodologi yang dilakukan adalah studi literatur, perancangan dan implementasi sistem, uji coba sistem, evaluasi, dan dokumentasi.

a. Studi Literatur

Pada tahap ini dilakukan pengumpulan informasi mengenai teori yang berkaitan dengan penelitian yaitu tentang pencurian identitas, kriptografi, steganografi, Least Significant Bit, kriptografi kurva eliptik, dan Portable Network Graphic.

b. Perancangan dan Implementasi Aplikasi

Pada tahap ini dilakukan perancangan dan pemrograman aplikasi yang disesuaikan dengan fungsionalitas dan spesifikasi yang telah ditentukan. Aplikasi yang dibangun merupakan aplikasi yang mengimplementasikan algoritma kriptografi kurva eliptik dan steganografi LSB. Kurva eliptik yang digunakan memiliki parameter yang akan menghasilkan 64 sebaran titik. Teknik LSB yang dipakai menggunakan teknik 1-*bit insertion*.

c. Uji Coba Aplikasi

Uji coba aplikasi dilakukan terhadap data berupa teks yang dimasukkan oleh *user* sesuai dengan *field* yang ada pada Kartu Tanda Penduduk (KTP) Indonesia dan juga gambar berformat PNG.

Pengujian algoritma dilakukan dengan tiga skenario, dimana masing-masing skenario akan diuji menggunakan parameter kurva yang berbeda. Data yang akan digunakan untuk masing-masing skenario adalah empat data dengan ukuran *cover image*, panjang data yang berbeda.

d. Evaluasi

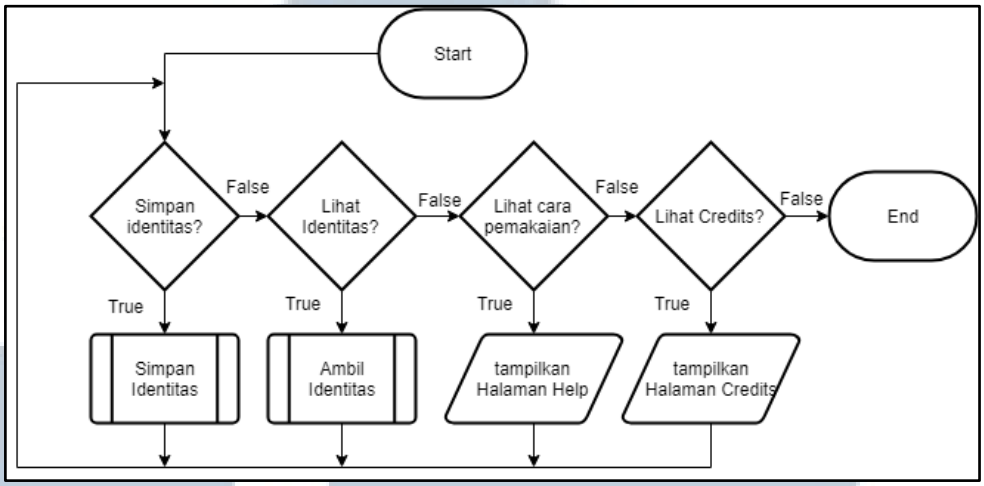
Evaluasi dilakukan untuk mengukur hasil keluaran aplikasi yaitu *file size* dari *stego-image*, dan nilai *Peak Signal to Noise Ratio (PSNR)* setelah disisipkan data terenkripsi yang dihitung menggunakan aplikasi PSNR 1.2, yang merupakan aplikasi *desktop* yang berfungsi untuk mengukur nilai PSNR dari dua gambar. Hasil evaluasi digunakan untuk mengetahui kekurangan dari aplikasi dan menentukan saran pengembangan untuk keperluan penelitian selanjutnya.

e. Dokumentasi

Dokumentasi dilakukan dari awal perancangan hingga kesimpulan akhir penelitian. Dokumentasi yang dilakukan berupa penulisan laporan.

3.2 Flowchart

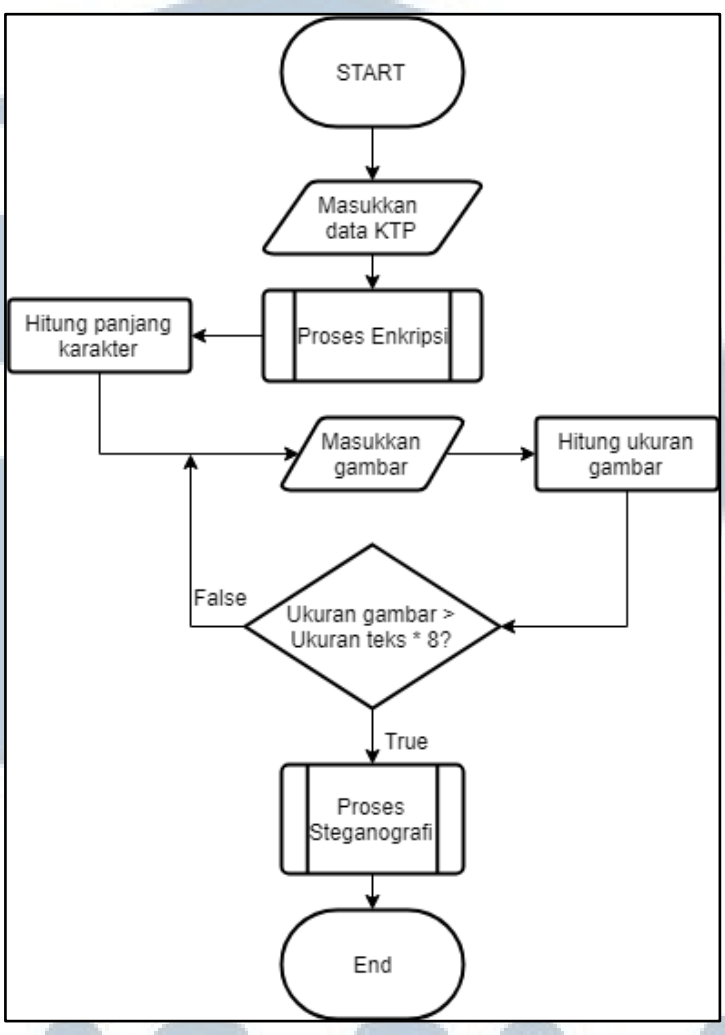
Pada aplikasi ini terdapat lima *flowchart* yang akan menjelaskan tentang alur kerja aplikasi. *Flowchart* tersebut menjelaskan tentang garis besar aplikasi, proses penyimpanan identitas, proses enkripsi, proses steganografi, proses melihat identitas, proses ekstrak teks, dan proses untuk dekripsi.



Gambar 3.1 *Flowchart* Aplikasi

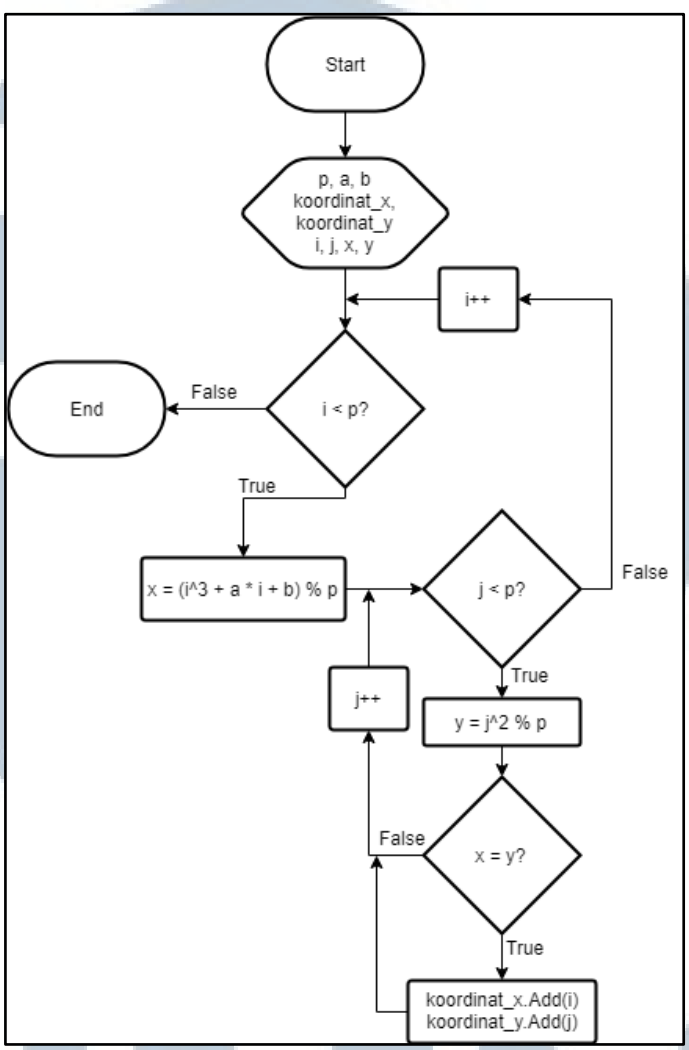
Aplikasi ini memiliki dua menu yaitu menu Simpan Identitas dan Lihat Identitas. Saat pertama kali aplikasi dijalankan, akan ditampilkan halaman untuk memilih menu.

Menu Simpan Identitas digunakan untuk mengenkripsi data dan menyimpan data tersebut ke dalam suatu gambar. Menu Lihat Identitas digunakan untuk mengambil data yang telah disimpan dalam gambar, lalu di dekripsi dan ditampilkan data yang sebenarnya.



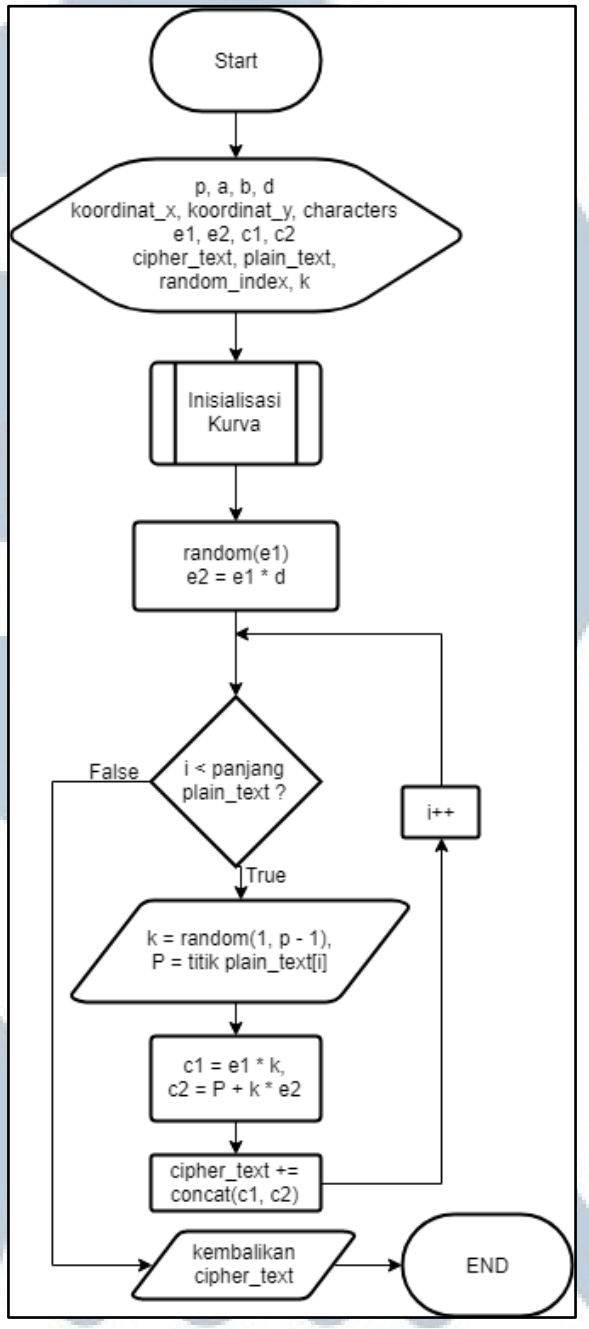
Gambar 3.2 *Flowchart* Simpan Identitas

Enkripsi diawali dengan melakukan pengisian data diri sesuai Kartu Tanda Penduduk (KTP) Indonesia. Data diri yang sudah dimasukkan akan dienkripsi sehingga menghasilkan data yang sudah terenkripsi, panjang data yang terenkripsi dihitung untuk menentukan ukuran gambar yang digunakan. *User* diminta mengunggah gambar untuk disisipkan data terenkripsi. Setelah itu akan dilakukan proses penyisipan data ke gambar.



Gambar 3.3 Proses Inisialisasi Kurva

Inisialisasi kurva digunakan untuk menentukan kurva yang akan digunakan dalam proses enkripsi dan dekripsi. Kurva yang berbeda akan menghasilkan titik-titik yang berbeda sehingga hasil enkripsi atau dekripsi pun akan berbeda untuk setiap kurvanya. Titik-titik dalam kurva ditentukan dengan mencari nilai *quaratic residue* yang mengacu pada Rumus 2.2, kemudian dibandingkan dengan hasil persamaan kurva yang mengacu pada Rumus 2.3. Jika hasilnya sama maka (i, j) merupakan titik dalam kurva dan disimpan ke dalam variabel yang telah disediakan. Setelah semua titik ditentukan, lalu dilanjutkan ke proses enkripsi *plain text*.

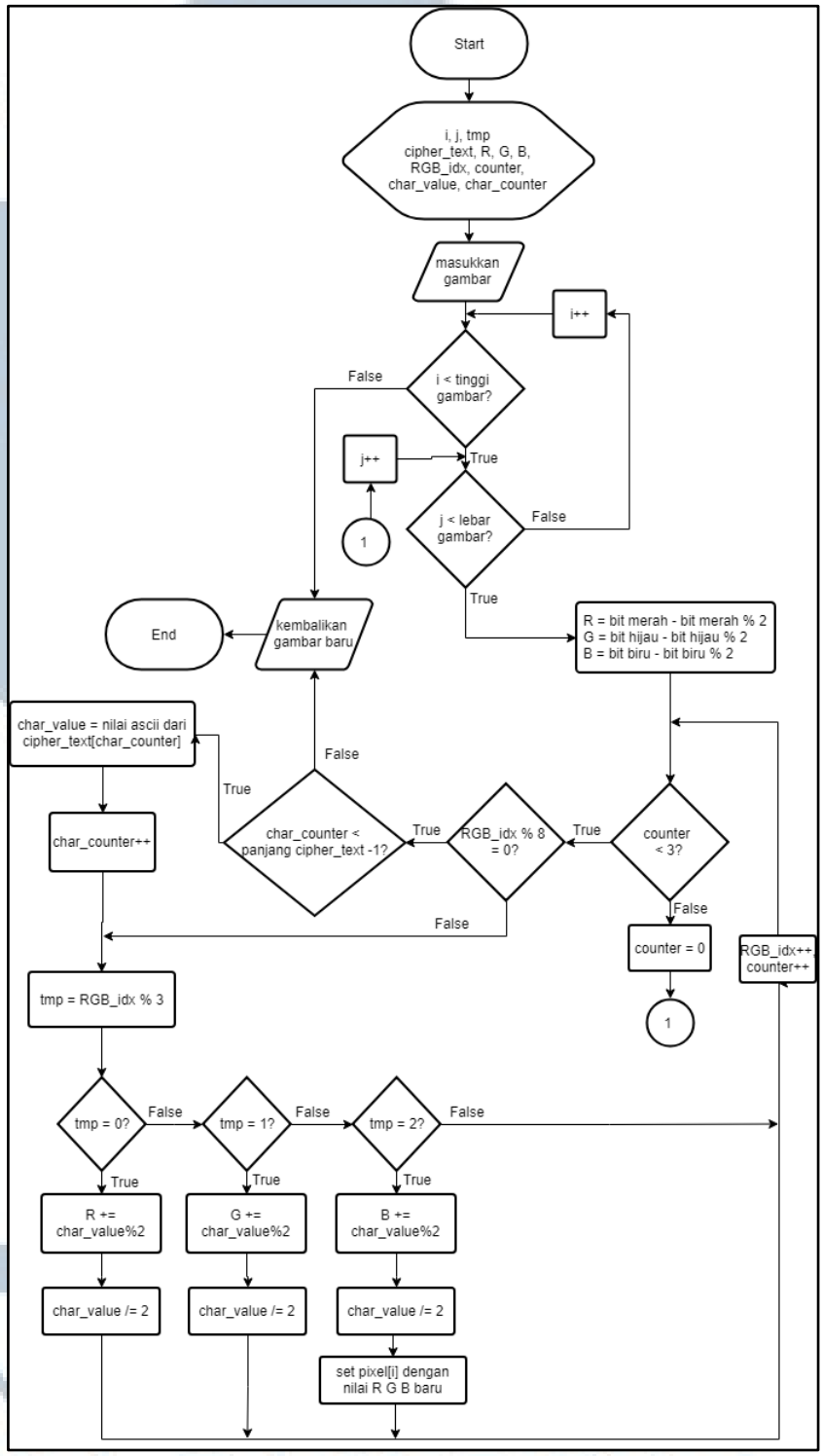


Gambar 3.4 Proses Enkripsi

Proses enkripsi diawali dengan inisialisasi kurva yang akan digunakan, variabel *d* merupakan bilangan riil yang dipilih secara acak sebagai *private key*. $e1$ dan $e2$ merupakan titik dalam kurva yang ditentukan sebagai *public key*, *koordinat_x*, *koordinat_y* digunakan untuk menampung titik-titik yang terdapat

dalam kurva. *plain_text* digunakan untuk menampung rangkaian karakter yang akan di enkripsi. P digunakan untuk menampung titik yang merepresentasikan karakter yang akan dienkripsi. $c1, c2$ merupakan *cipher* dari tiap karakter. *cipher_text* digunakan untuk menampung rangkaian *cipher*.

Iterasi dilakukan sebanyak panjang *plain text* yang akan dienkripsi. Untuk setiap karakter yang akan dienkripsi, dipilih satu bilangan acak k. Titik yang merepresentasikan karakter tersebut (P) akan dihitung untuk menghasilkan *cipher text*. Setiap karakter pada *plain text* akan menghasilkan dua karakter *cipher text* yaitu $c1$ dan $c2$. $c1$ didapatkan dari hasil kali antara $e1$ dengan k, yang mengacu pada Rumus 2.8 sedangkan $c2$ didapatkan dari hasil kali antara $e2$ dengan k lalu dijumlahkan dengan P, yang mengacu pada Rumus 2.9. Operasi $c1$ dan $c2$ merupakan operasi aritmatika titik seperti penggandaan titik, penjumlahan titik, pengurangan titik dan perkalian titik. Hasil $c1$ dan $c2$ tersebut kemudian akan digabungkan menjadi *cipher text* untuk satu karakter, sehingga setiap katakter yang dienkripsi akan menghasilkan *cipher text* berupa $c1c2$.



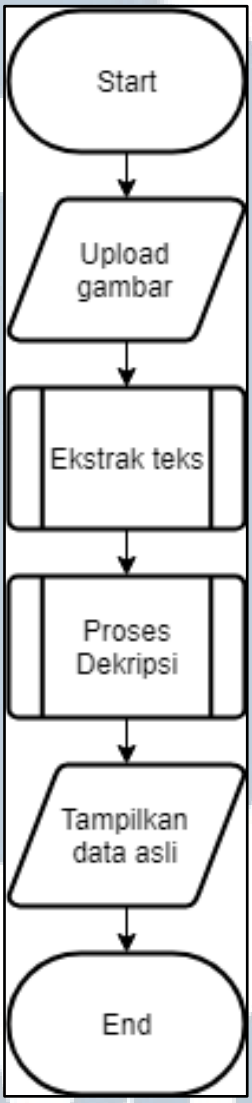
Gambar 3.5 Proses Steganografi

Proses steganografi diawali dengan melakukan inisialisasi variabel *cipher_text* untuk menampung data yang disembunyikan dalam gambar. Variabel

R, *G*, dan *B* yang masing-masing akan menampung bit warna merah, hijau, dan biru dari tiap *pixel*. *RGB_idx* digunakan untuk menentukan pada warna apa *bit* data harus disembunyikan. *counter* digunakan untuk mengecek apakah pada satu *pixel* masih bisa digunakan untuk menampung data atau tidak. *char_value* digunakan untuk menampung nilai ASCII dari karakter yang akan disembunyikan. *char_counter* digunakan untuk mengambil satu karakter pada *cipher_text*.

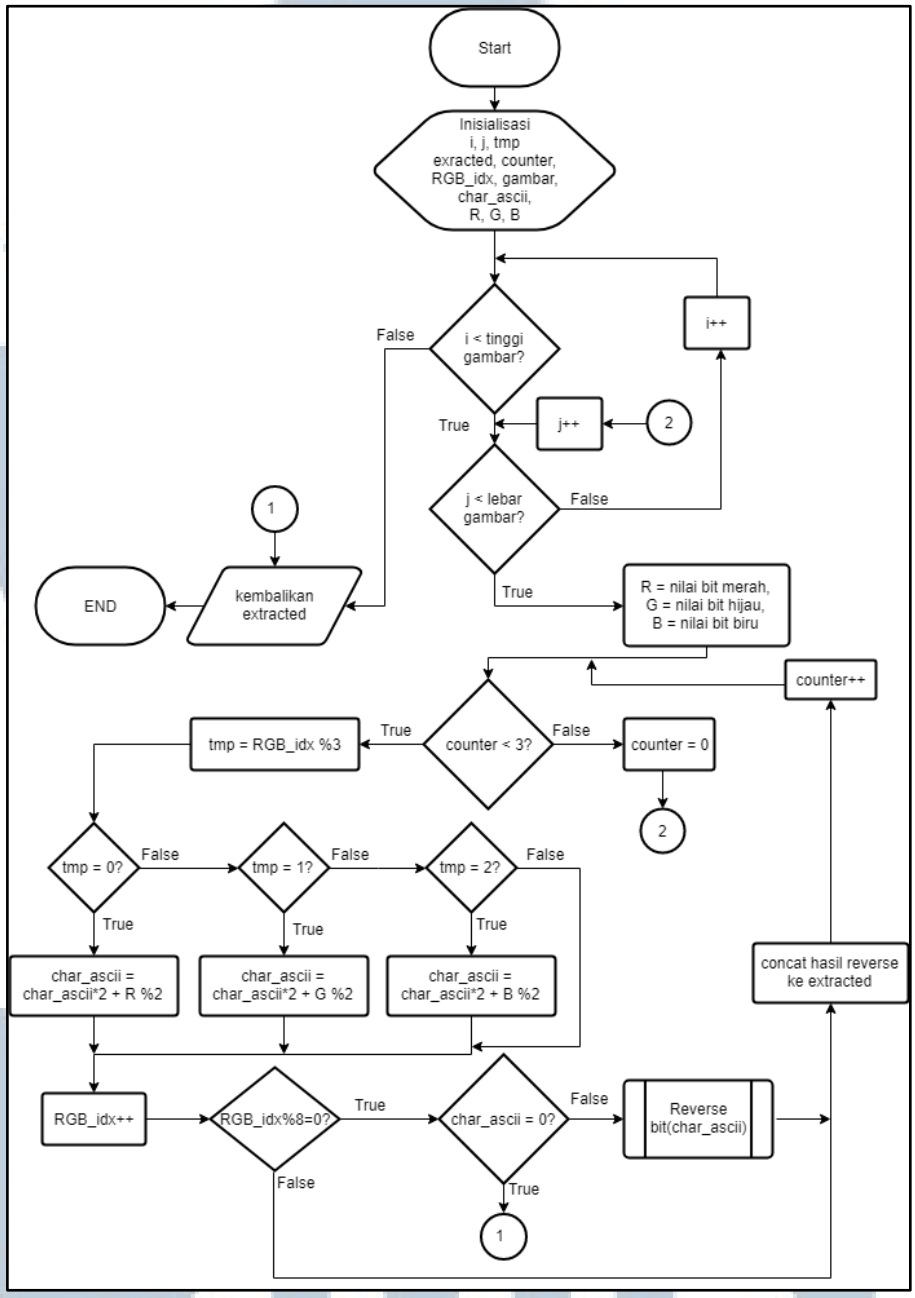
Ambil *pixel* dari gambar, dan untuk setiap *pixel* kosongkan *bit* paling kanan dari tiap elemen warna. Karena satu *pixel* memiliki tiga elemen warna, maka satu *pixel* dapat menampung tiga *bit* data. Operasi $RGB\_idx \bmod 8$ digunakan untuk menentukan apakah seluruh *bit* pada suatu karakter sudah disembunyikan, jika sudah maka akan diambil karakter baru untuk disembunyikan.

Untuk tiap *bit* pada karakter yang akan disembunyikan, akan ditentukan apakah *bit* tersebut disembunyikan ke dalam elemen warna merah, hijau, atau biru. Setelah tiga elemen warna dimanipulasi dengan *bit* karakter, maka disimpan kembali kedalam *pixel* dengan nilai baru.



Gambar 3.6 *Flowchart* Ambil Identitas

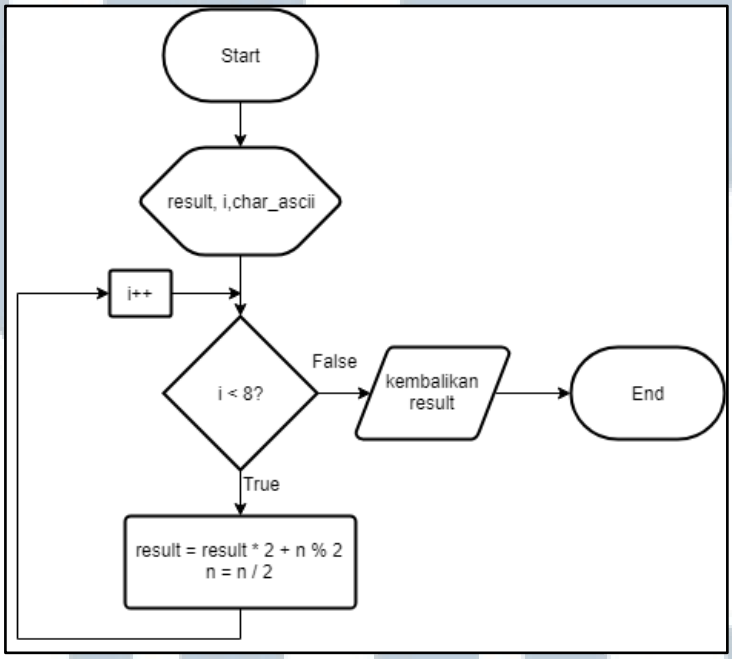
Proses untuk melakukan dekripsi dapat dilihat seperti pada Gambar 3.6. Pertama, gambar yang berisi data terenkripsi diunggah terlebih dahulu. Setelah itu akan dilakukan proses untuk mengekstrak data tersebut dari gambar. Data yang sudah diekstrak berbentuk data teks yang terenkripsi, sehingga dilanjutkan proses dekripsi. Setelah itu baru ditampilkan data diri yang sebenarnya.



Gambar 3.7 *Flowchart* Ekstrak Teks

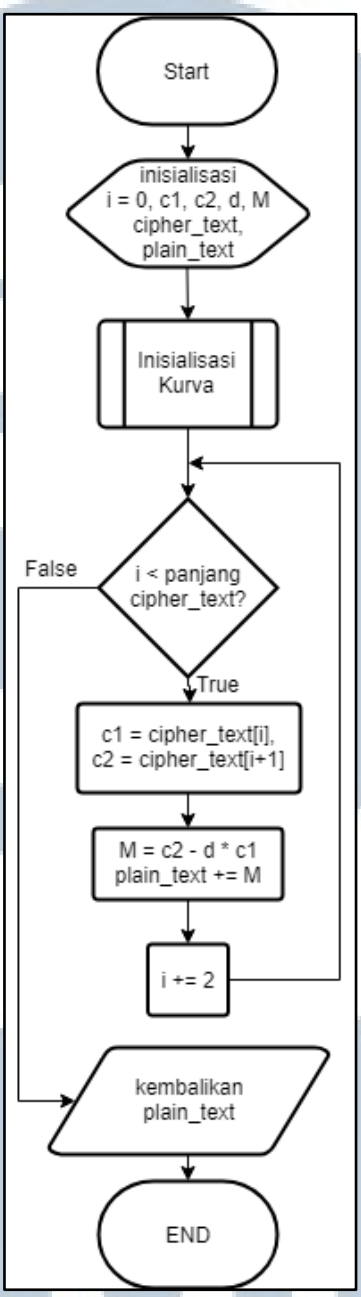
Proses pengambilan *bit* data dilakukan dengan melakukan iterasi tiap *pixel* dan mengambil *bit* paling kanan dari tiap elemen warna pada tiap *pixel*. *Bit* paling kanan dari tiap elemen warna akan ditampung ke dalam variabel *char_value*. Proses pengambilan *bit* dilakukan dengan cara menghitung operasi modulus dua pada tiap

nilai elemen warna. Jika sudah didapatkan nilai delapan *bit* pada variabel *char_value* maka nilai tersebut dapat diubah menjadi karakter. Untuk merubah menjadi karakter, nilai bit yang telah didapatkan perlu dibalik susunannya untuk baru kemudian dimasukkan ke dalam variabel *extracted* yang berfungsi untuk menampung data teks yang sudah diekstrak.



Gambar 3.8 Proses *Reverse Bit*

Proses *Reverse Bit* digunakan untuk membalik susunan bit yang telah kita dapatkan. Proses ini diperlukan karena proses penyisipan dimulai dari *bit* paling belakang, sehingga saat dilakukan proses ekstraksi, data yang terambil lebih dulu adalah *bit* paling belakang dari data sebenarnya. Maka dari itu untuk didapatkan nilai yang sebenarnya perlu dilakukan penyusunan ulang.

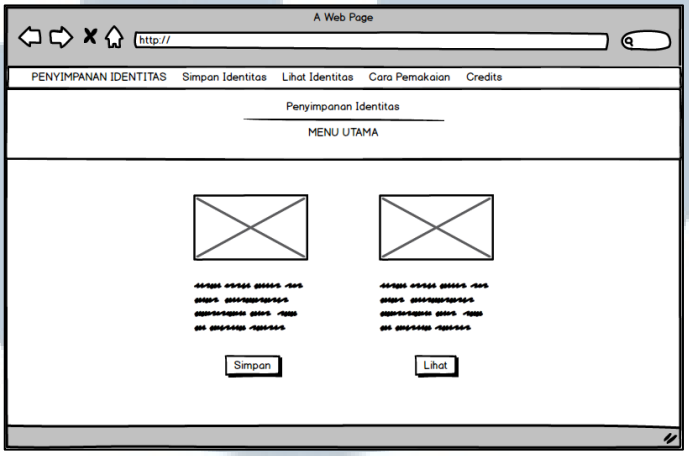


Gambar 3.9 Proses Dekripsi

Setiap karakter pada data asli dienkripsi menjadi dua karakter baru. Maka proses dekripsi dilakukan dengan mengambil dua karakter terenkripsi yaitu $c1$ dan $c2$ secara berurutan. Karakter asli (M) dapat diambil dengan menggunakan Rumus 2.10. Proses ini diulang sampai semua karakter *cipher text* selesai didekripsi.

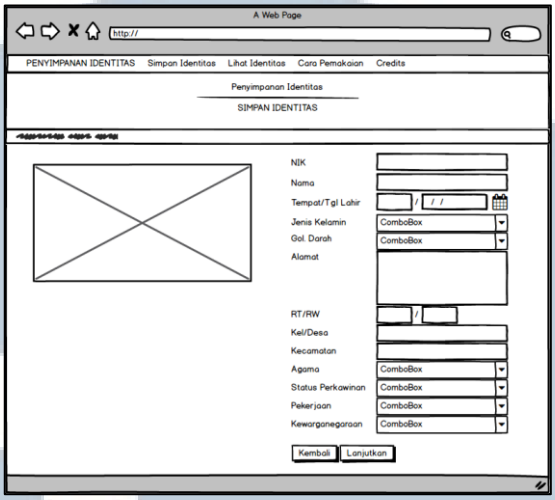
Setelah didapatkan karakter asli, maka karakter tersebut dimasukkan ke dalam suatu variabel penampung. Jika semua karakter *cipher text* telah diproses maka variabel penampung akan berisi data yang sebenarnya.

3.3 Rancangan Tampilan Antarmuka



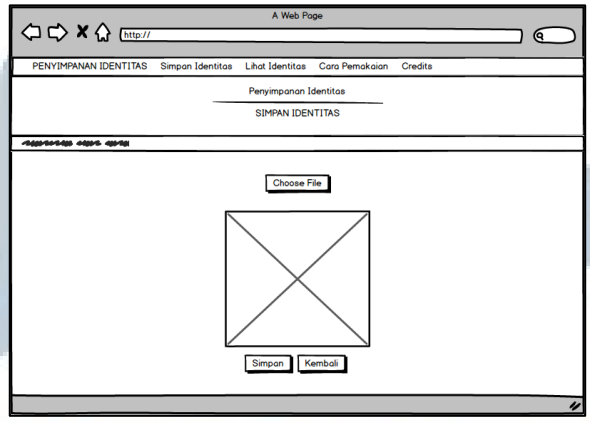
Gambar 3.10 Halaman Awal Aplikasi

Gambar 3.10 merupakan halaman yang ditampilkan saat pertama kali aplikasi dijalankan. Terdapat dua tombol menu untuk memilih proses mana yang ingin dilakukan. Tombol Simpan Identitas akan memanggil proses Simpan Identitas sedangkan tombol Lihat Identitas akan memanggil proses Lihat Identitas.



Gambar 3.11 Rancangan Halaman Simpan Identitas

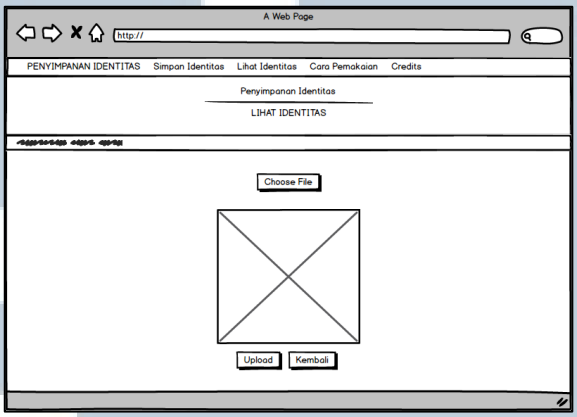
Gambar 3.11 menunjukkan rancangan halaman untuk memasukkan data diri yang sesuai pada Kartu Tanda Penduduk (KTP) Indonesia yang ingin diarsipkan. Setiap kolom *input* dibuat mengikuti *field* pada KTP. Tombol Lanjutkan digunakan untuk melanjutkan ke proses pengunggahan foto untuk menampung data diri tersebut.



Gambar 3.12 Rancangan Halaman *Upload* Gambar

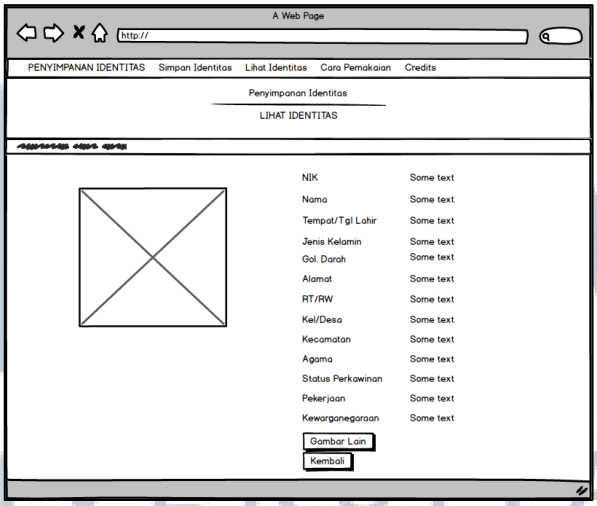
Gambar 3.12 menunjukkan halaman setelah mengisi data diri. Pada halaman ini akan ditampilkan ukuran gambar minimal yang dapat digunakan untuk menyimpan data diri. Semakin panjang data diri yang disimpan maka semakin besar

ukuran gambar yang dibutuhkan. Tombol Simpan digunakan untuk melakukan proses penyisipan data ke dalam gambar yang dipilih.



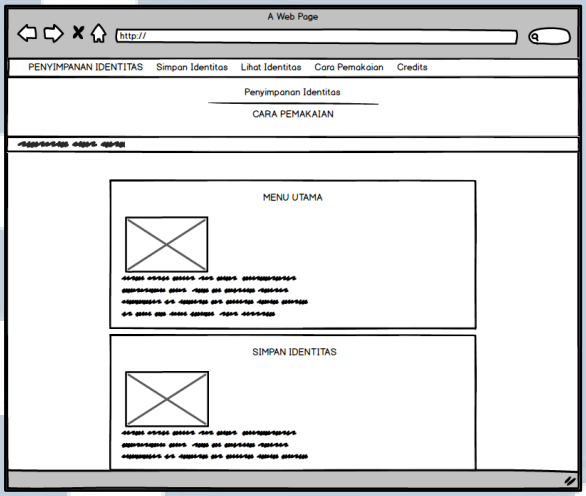
Gambar 3.13 Rancangan *Upload Stego Image*

Gambar 3.13 menunjukkan halaman untuk melihat identitas dari gambar yang dipilih. Tombol Lihat Identitas digunakan untuk melakukan proses ekstrak teks dan dekripsi.



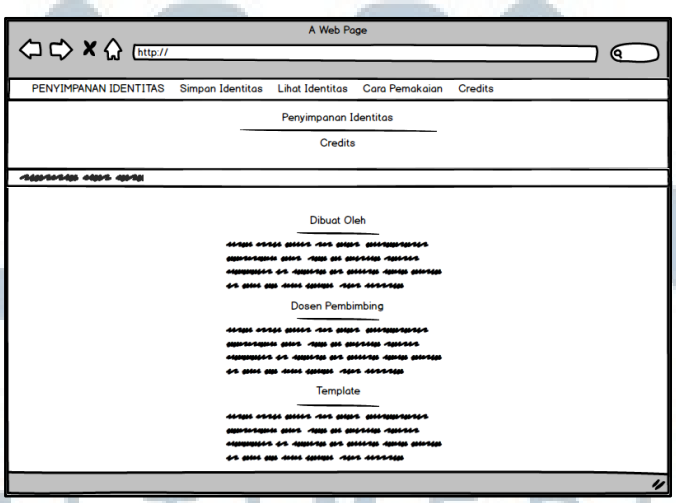
Gambar 3.14 Rancangan Halaman Lihat Identitas

Gambar 3.14 menunjukkan halaman yang menampilkan data yang sudah diambil dan didekripsi dari gambar. Data akan ditampilkan pada kolom sesuai dengan kolom pada halaman Simpan Identitas.



Gambar 3.15 Rancangan Halaman Cara Pemakaian

Gambar 3.15 menunjukkan halaman untuk menjelaskan cara pemakaian aplikasi. Terdapat tiga bagian, yaitu untuk menjelaskan sekilas tentang aplikasi, cara menyimpan identitas, dan cara melihat identitas yang telah disimpan.



Gambar 3.16 Rancangan Halaman *Credits*

Gambar 3.16 menunjukkan halaman yang berisi tentang setiap orang, *libraries, template* yang yang memiliki andil dalam pembangunan aplikasi.