BAB III
ANALISA DAN PERANCANGAN

3.1 Metode Penelitian

Dalam melakukan penelitian tugas akhir ini, digunakan metode studi literatur dan eksperimen dengan langkah-langkah sebagai berikut.

3.1.1 Studi Literatur

Melakukan studi mengenai referensi-referensi yang berhubungan dengan algoritma VEA, algoritma AES, enkripsi video, dan video *streaming* untuk mendapatkan pengertian dasar mengenai metode dan cara-cara membangun aplikasi ini referensi tersebut dapat berupa jurnal, artikel, buku, dan lain-lain.

3.1.2 Analisa Aplikasi

Mengidentifikasi kendala yang akan dihadapi dalam proses pembangunan aplikasi serta menentukan batasan-batasan dan kebutuhan yang diperlukan serta merancang piranti lunak untuk optimasi *Video Encryption Algorithm* dengan *Advanced Encryption Algorithm* pada *Securd Streaming*.

3.1.3 Perancangan Aplikasi

Setelah melakukan analisa aplikasi, proses selanjutnya adalah merancang aplikasi berdasarkan hasil analisis. Perancangan dilakukan dengan merancang *user interface* dan membuat diagram-diagram seperti *Entity Relationship Diagram*, dan *Flow Chart*.

3.1.4 Pembangunan Aplikasi

Aplikasi dibuat berdasarkan diagram-diagram dan rancangan *user interface* yang telah dibuat sebelumnya.

3.1.5 Pengujian dan Evaluasi

Melakukan pengujian proses enkripsi video dari hasil optimasi algoritma VEA pada video berformat *MPEG4-MP4*. Lalu dilakukan evaluasi dengan membandingkan kualitas asli video dengan video yang telah melalui proses dekripsi untuk mengetahui adanya perubahan kualitas video yang telah melalui

proses enkripsi. Melakukan perhitungan waktu dibutuhkan untuk melakukan proses upload dan enkripsi.

3.2 Spesifikasi Perancangan Sistem

Spesifik perancangan sistem menerangkan pembuatan sistem seperti fungsionalitas aplikasi, batasan aplikasi, masukan aplikasi, dan keluaran aplikasi**.**

3.1.1 Fungsionalitas Aplikasi

Aplikasi yang dibuat memiliki fungsionalitas sebagai berikut.

1. Membaca *file MPEG4-MP4* yang akan digunakan.
2. Melakukan proses enkripsi dengan algoritma *Video Encryption Algorithm* yang dioptimasi dengan algoritma *Advanced Encryption Algorithm*.
3. Menghasilkan keluaran *file* video dan *file* kunci yang telah terenkripsi.
4. Menghasilkan keluaran file yang telah didekripsi yang kemudian akan di-*streaming* melalui *web-browser*.

3.1.2 Masukan dan Keluaran Aplikasi

Masukan yang dibutuhkan sistem berupa :

1. Lokasi penyimpanan dari *file* video yang akan digunakan untuk dienkripsi dan didekripsi.
2. Lokasi dan nama *file* yang telah dilakukan proses enkripsi dan dekripsi.
3. Kata sandi yang digunakan untuk melakukan proses enkripsi dan dekripsi

Keluaran dari sistem berupa :

1. Lama waktu proses enkripsi atau dekripsi
2. *File* hasil enkripsi atau dekripsi

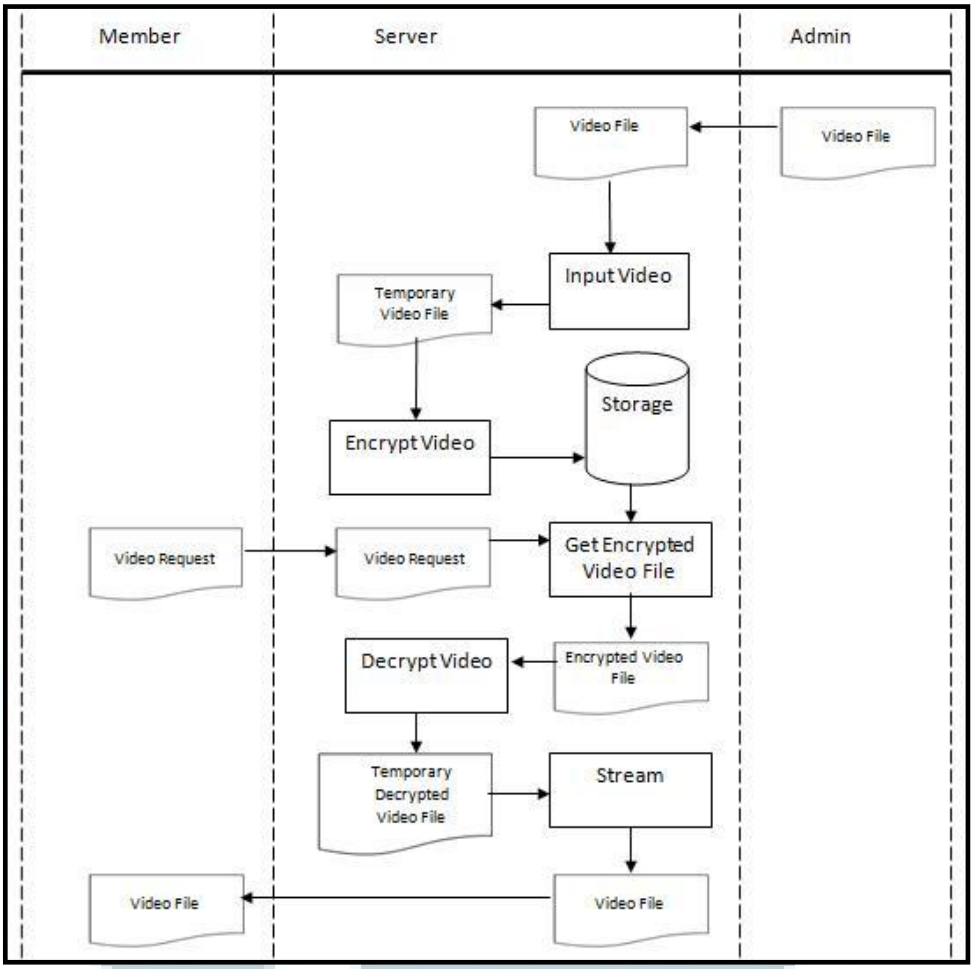
3.3 Analisa Pengembangan Program

Berdasarkan teori yang dipaparkan sebelumnya, algoritma VEA merupakan algoritma dengan waktu enkripsi pendek. Namun, memiliki tingkat keamanan yang relatif rendah (Seidel, Socek, & Sramka, 2005). Proses enkripsi

akan dilakukan pada video MPEG. Untuk meningkatkan tingkat keamanannya, maka VEA akan dimodifikasi dengan menambahkan algoritma AES. Algoritma AES digunakan karena komputasinya yang ringan dan merupakan kriptografi yang memiliki tingkat keamanan lebih baik daripada DES (Surian, 2006).

Pada optimasi algoritma VEA yang dikembangkan ini, diusulkan kunci yang digunakan untuk enkripsi dengan algoritma VEA yang sudah dienkripsi menggunakan algoritma AES. Penambahan algoritma AES untuk mengenkripsi kunci yang akan digunakan ini menjadi perbedaan antara algoritma VEA yang telah dioptimasi dengan VEA pada umumnya.

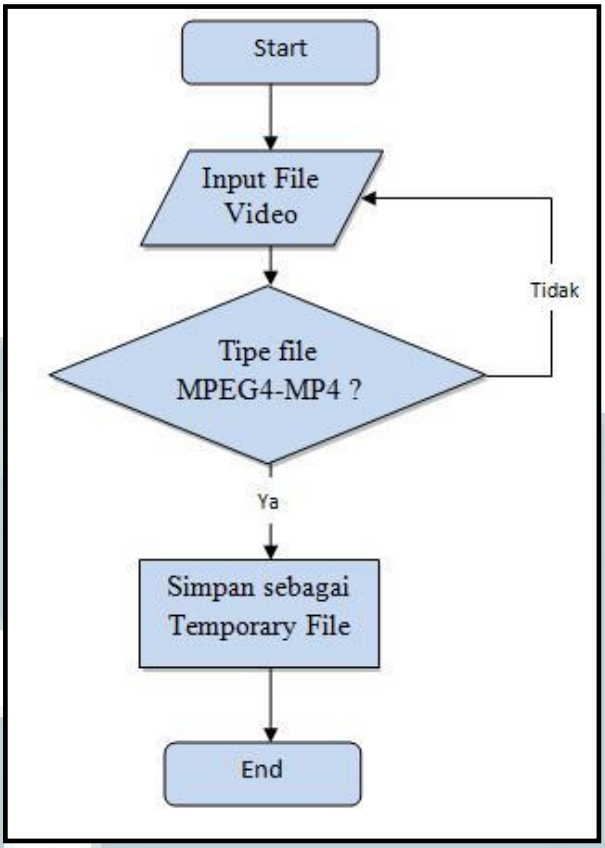
Algoritma AES yang ada digunakan pada program ini dibuat dengan fitur yang telah ada di PHP. Optimasi yang dilakukan adalah membuat kunci acak yang dienkripsi dengan AES dengan tujuan untuk meningkatkan keamanan dengan membuat kunci yang berbeda-beda untuk setiap enkripsi.



Gambar 3.1 System Flow Aplikasi

Gambar 3.1 diatas menunjukkan *system flow* aplikasi, admin akan melakukan *input* data video ke server. Proses "Input Video" akan mengubah *file* video menjadi *temporary file*. Kemudian, proses "Encrypt Video" akan mengenkripsi *temporary file* yang kemudian akan disimpan di *storage*.

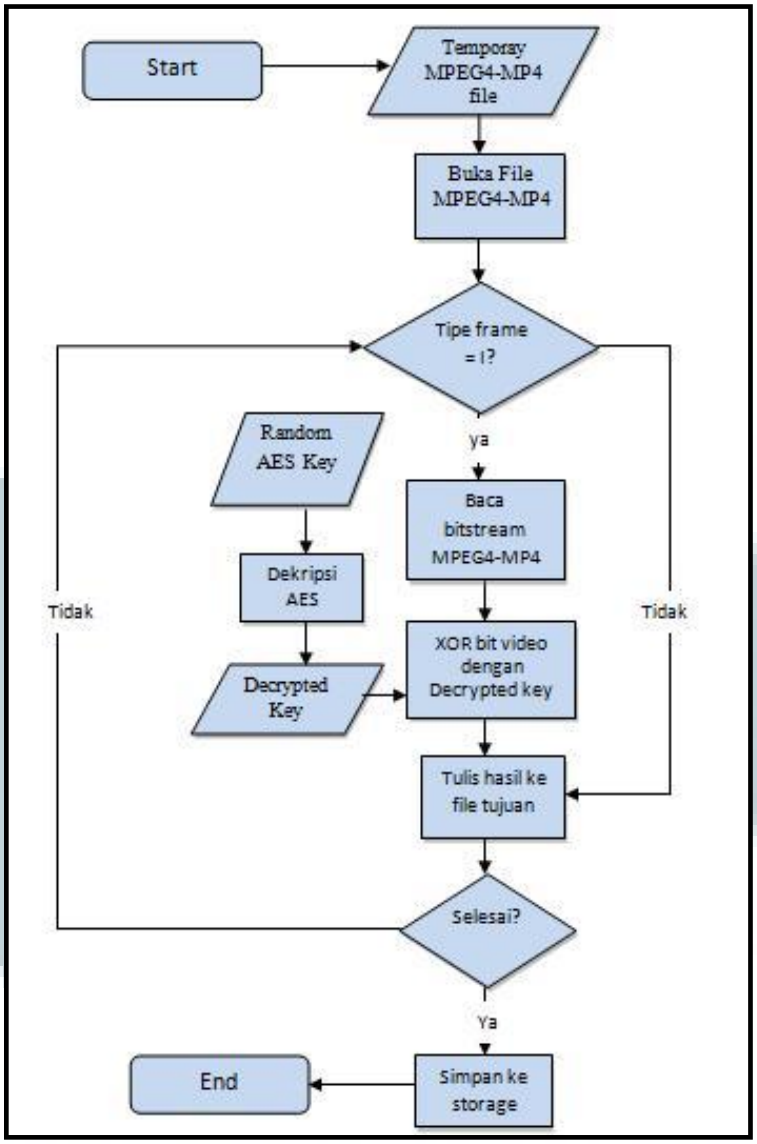
Apabila *member* melakukan *request* untuk menonton video yang ada di dalam *storage*, maka *request* akan di proses oleh "Get Encrypted Video File" yang akan mengambil *file* video di *storage*. Kemudian, *file* video yang masih terenkripsi akan didekripsi oleh proses "Decrypt Video". Proses "Decrypt Video" akan menghasilkan file *temporary* yang akan digunakan oleh proses "Stream". Kemudian proses "Stream" akan menampilkan video yang dapat dilihat oleh *member*.



Gambar 3.2 Diagram Alir Proses Input Video

Gambar 3.2 merupakan diagram alir proses input video berjalan. Berikut adalah penjelasan proses bagaimana *input* video berjalan :

1. Tipe *file* video yang dimasukkan pengguna akan diperiksa.
2. Jika *file* video bukan MPEG4-MP4, maka pengguna akan diminta untuk memasukkan *file* video dengan tipe MPEG4-MP4.
3. Jika file video bertipe MPEG4-MP4, maka file akan disimpan sebagai *temporary file*.

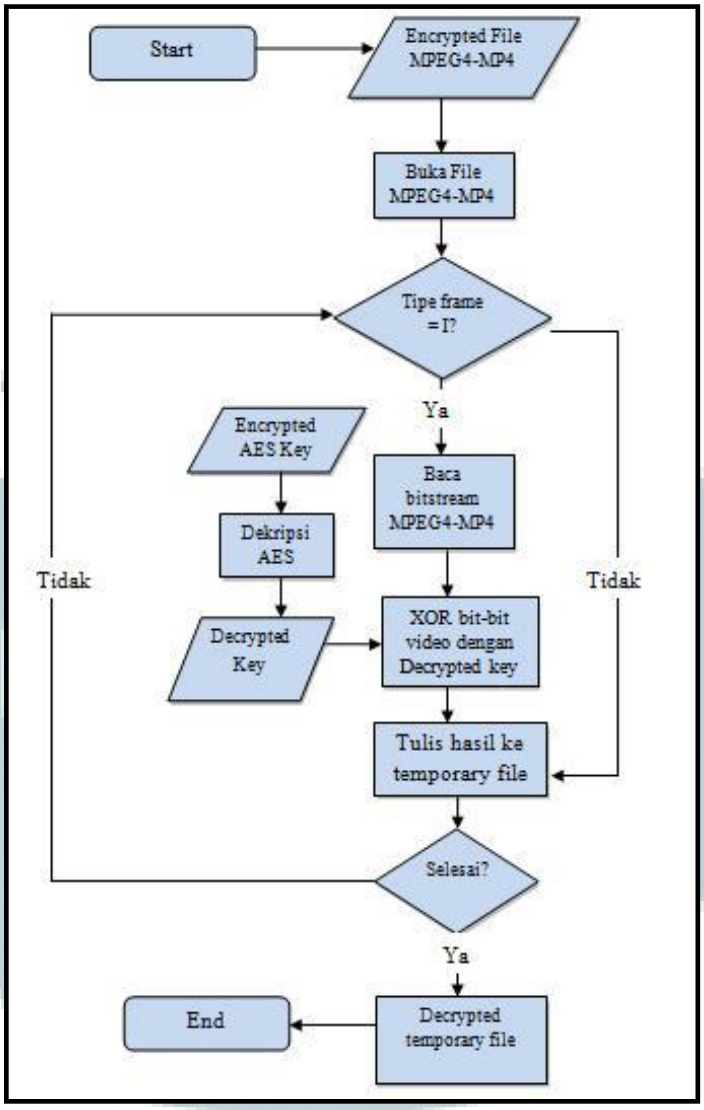


Gambar 3.3 Diagram Alir Optimasi VEA dengan AES (Enkripsi)

Gambar 3.3 diatas merupakan diagram alir bagaimana proses enkripsi dengan algoritma VEA yang telah dioptimasi bekerja. Berikut adalah proses bagaimana algoritma VEA yang telah dioptimasi dengan algoritma AES berjalan:

1. Buka *temporary file* MPEG4-MP4.
2. Baca *frame file* MPEG4-MP4, baca tipe *frame*-nya.
3. Baca *stream bit* dari *frame* tersebut.
4. Jika *frame* dari stream bit bukan *frame* I, maka *stream* bit langsung ditulis ke file tujuan.
5. Jika stream bit tersebut merupakan *stream* bit dari *frame* I, maka bit-bit tersebut di-XOR-kan dengan kunci.

6. Kunci yang digunakan merupakan kunci yang telah dibuat secara acak dan terenkripsi dengan algoritma AES dan akan didekripsi agar dapat digunakan.
7. Hasil XOR *stream* bit dengan kunci yang merupakan hasil enkripsi ditulis ke *file* tujuan.
8. Baca *frame* selanjutnya, kembali ke langkah nomor 2 sampai *End-of-File*.
9. Setelah dibaca sampai *End-of-File*, hasil enkripsi akan disimpan di *storage*.

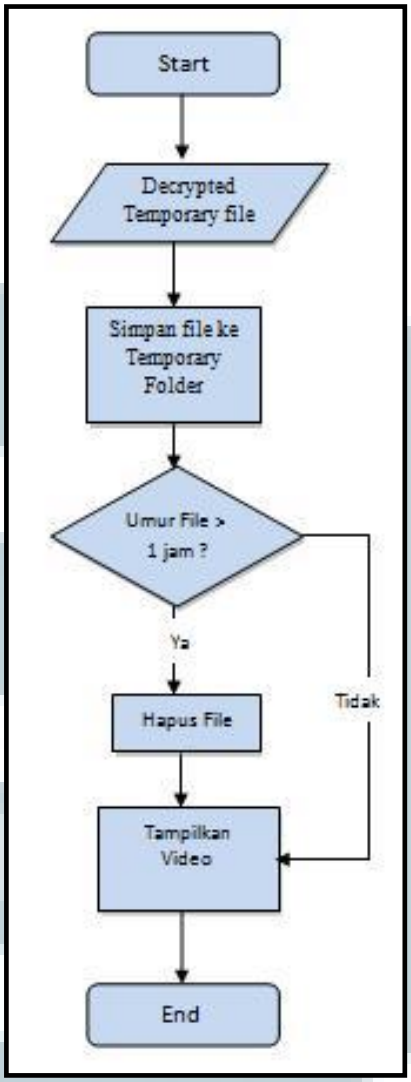


Gambar 3.4 Diagram Alir Optimasi VEA dengan AES (Dekripsi)

Gambar 3.4 merupakan diagram alir proses dekripsi dengan menggunakan algoritma VEA yang telah dioptimasi dengan algoritma AES. Berikut adalah proses bagaimana algoritma VEA yang telah dioptimasi dengan algoritma AES berjalan:

1. Buka file video MPEG.
2. Baca *frame file* MPEG, baca tipe *frame*-nya.
3. Baca *stream bit* dari *frame* tersebut.
4. Jika *frame* dari stream bit bukan frame I, maka *stream* bit langsung ditulis ke file tujuan.

5. Jika stream bit tersebut merupakan *stream* bit dari *frame* I, maka bit-bit tersebut di-XOR-kan dengan kunci.
6. Kunci yang digunakan terenkripsi dengan algoritma AES dan akan didekripsi agar dapat digunakan
7. Hasil XOR *stream* bit dengan kunci yang merupakan hasil enkripsi ditulis ke *file temporary* yang akan otomatis terhapus setiap satu jam.
8. Baca *frame* selanjutnya, kembali ke langkah nomor 2 sampai *End-of-File*. Kemudian akan digunakan untuk *streaming.*

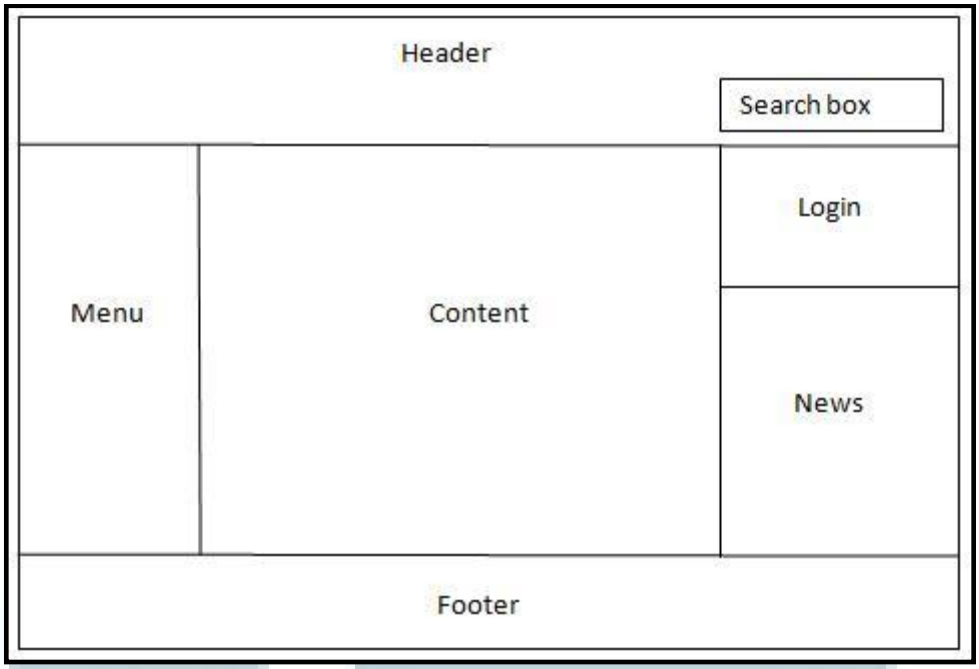


Gambar 3.5 Diagram Alir Proses Streaming

Gambar 3.5 merupakan diagram alir proses streaming akan berjalan setelah proses dekripsi menghasilkan *output file* video yang telah didekripsi.. Berikut adalah proses bagaimana proses streaming berjalan:

1. Simpan *file* video yang telah didekripsi ke *temporary* folder.
2. Cek *file* yang ada di dalam folder *temporary*.
3. Jika ada *file* berumur lebih dari satu jam, maka *file* tersebut dihapus
4. Jika tidak ada *file* berumur lebih dari satu jam, maka *file* video akan langsung ditampilkan.

3.4 Desain Sistem



Gambar 3.6 Sketsa Layar

Dalam desain sistem ini dipaparkan mengenai tampilan antarmuka yang dibuat. Gambar 3.6 diatas adalah sketsa layar secara umum, dimana layar terbagi menjadi empat bagian, yaitu bagian *sidebar* menu, bagian *sidebar login*, bagian *sidebar* news, bagian *search*, bagian *header*, bagian *footer*, bagian *content*. Dari seluruh bagian hanya bagian *content*, *sidebar* menu dan *sidebar login* yang tampilannya dapat berubah.

Gambar 3.7 Sketsa *Sidebar* Menu

Gambar 3.7 diatas adalah sketsa dari *sidebar menu* sebelum *user* melakukan login. Apabila *user* melakukan login sebagai *admin*. Maka *sidebar menu* akan berubah seperti gambar 3.8.

Gambar 3.8 Sketsa *sidebar menu* admin

Gambar 3.8 diatas adalah sketsa dari *sidebar menu* setelah *user* melakukan login sebagain *admin.* Menu yang ditandai dengan warna merah adalah menu untuk *admin.* Secara garis besar, Sistem Seni Digital bagian admin dibagi menjadi 4 menu utama, dengan pembagian sebagai berikut.

1. Home

Pada menu Home terdapat dua sub-menu untuk *admin*, dengan pembagian sebagai berikut.

a. News

Pada bagian News user dapat menambah, mengubah, dan menghapus berita yang akan ditampilkan di Seni Digital.

b. Language

Pada bagian language user dapat menambah, mengubah, dan menghapus daftar bahasa.

2. Animation

User dapat melihat daftar video dengan tipe *animation.* Selain itu, user dapat menghapus dan mengubah video yang ada dalam daftar. Pada bagian Animation terdapat 4 sub-menu, dengan pembagian sebagai berikut.

a. Most View

Menampilkan 10 video tipe *animation* yang paling banyak dilihat. Di sub-menu ini *user* dapat mengubah dan menghapus video yang ada di dalam tabel.

b. Latest View

Menampilkan 10 video tipe *animation* yang terakhir dilihat. Di sub-menu ini *user* dapat mengubah dan menghapus video yang ada di dalam tabel

c. Add Animation

User dapat mengunggah video dengan tipe *animation*.

d. Manage Genre

Ketika mengklik "Manage Genre", akan ditampilkan daftar genre. *User* dapat menambah, menghapus dan mengubah daftar genre animation.

3. Movie

User dapat melihat daftar video dengan tipe *movie*. Selain itu, user dapat menghapus dan mengubah video yang ada dalam daftar. Pada bagian Movie terdapat 4 sub-menu, dengan pembagian sebagai berikut.

a. Most View

Menampilkan 10 video tipe *animation* yang paling banyak dilihat. Di sub-menu ini *user* dapat mengubah dan menghapus video yang ada di dalam tabel.

b. Latest View

Menampilkan 10 video tipe *animation* yang terakhir dilihat. Di sub-menu ini *user* dapat mengubah dan menghapus video yang ada di dalam tabel

c. Add Movie

User dapat mengunggah video dengan tipe *movie*.

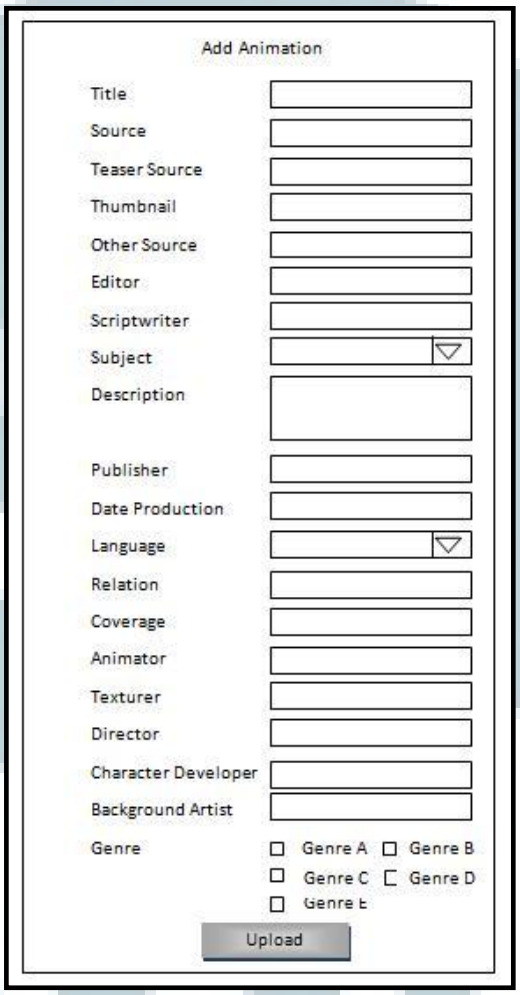
d. Manage Genre

Ketika mengklik "Manage Genre", akan ditampilkan daftar genre. *User* dapat menambah, menghapus dan mengubah daftar genre movie.

4. About

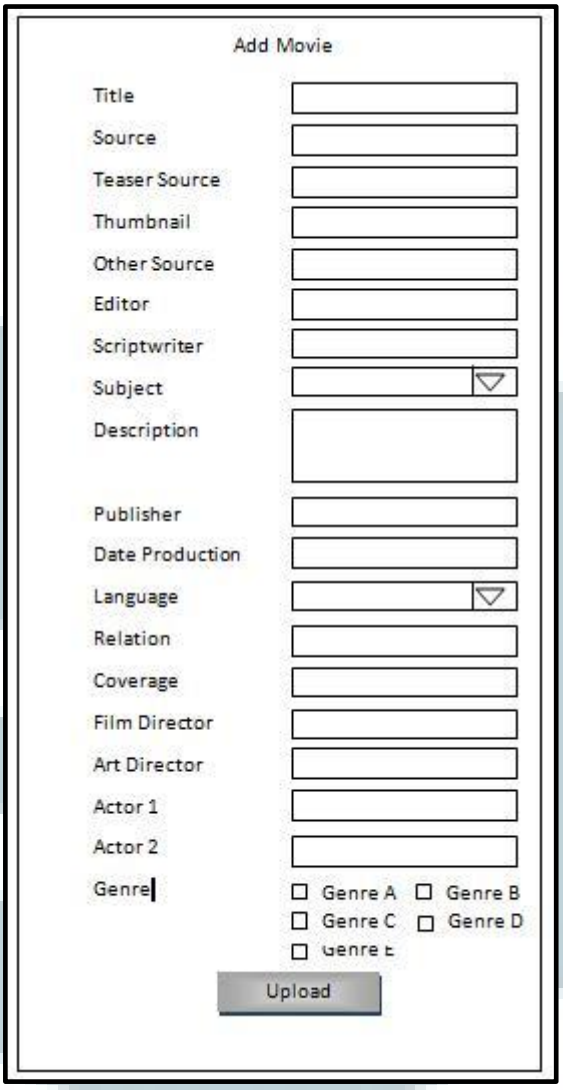
User dapat melihat informasi mengenai Seni Digital Universitas Multimedia Nusantara.

Pada laporan ini hanya akan dibahas mengenai sub-menu yang berhubungan dengan enkripsi video, yaitu Add Animation, Add Movie, Edit Animation dan Edit Movie.



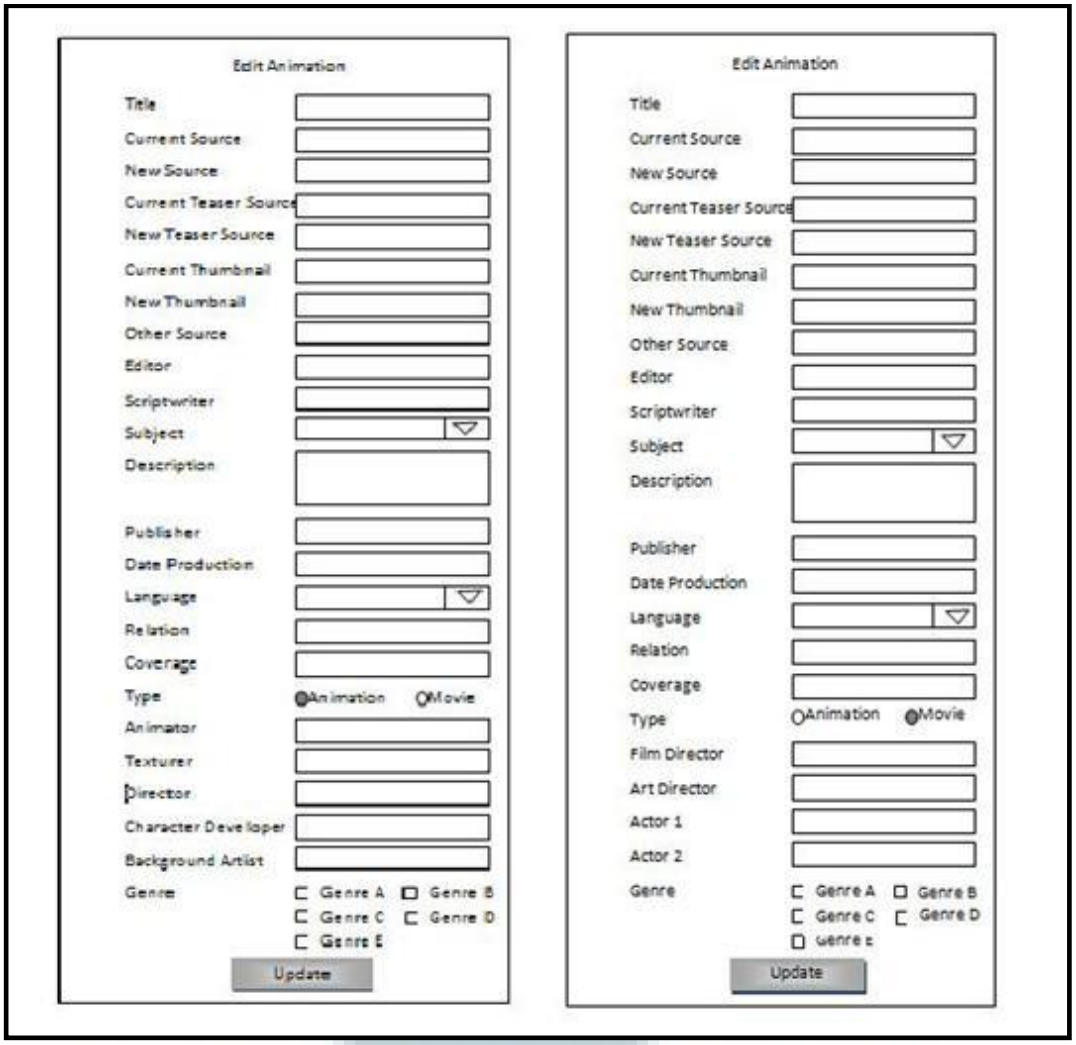
Gambar 3.9 Sketsa Add Animation

Gambar 3.9 diatas adalah sketsa dari halaman *Add Animation* dan hanya muncul apabila *user login* sebagai *admin.* Halaman ini muncul apabila *user* mengklik sub-menu "Add Animation" yang ada di halaman Animation. Untuk mengunggah, *admin* wajib mengisi keterangan di field *title, thumbnail, editor, description, publisher, date production, language, animator, genre,* dan *source.* Video yang dapat diunggah adalah video dengan format MPEG4-MP4.



Gambar 3.10 Sketsa Add Movie

Gambar 3.10 diatas adalah sketsa dari halaman *Add Movie* dan hanya muncul apabila *user login* sebagai *admin* . Halaman ini muncul apabila *user* mengklik sub-menu "Add Movie" yang ada di halaman Movie. Untuk mengunggah, a*dmin* wajib mengisi keterangan di field *title ,thumbnail, editor, description, publisher, date production, language, film director, genre,* dan *source.* Video yang dapat diunggah adalah video dengan format MPEG4-MP4.



Gambar 3.11 Sketsa Edit Animation dan Edit Movie

Gambar 3.11 diatas adalah sketsa dari halaman *Edit Movie,* halaman ini muncul apabila *user* mengklik "Edit" yang ada di tabel pada halaman Movie. Pada halaman ini user dapat mengubah informasi mengenai video. Apabila *user* mengganti keterangan yang ada di field *type* menjadi *animation*, maka field-field yang berada dibawah field *type* akan berubah menjadi field-field untuk video animation.