

BAB 3 METODOLOGI PENELITIAN

3.1 Metodologi Penelitian

Metodologi yang digunakan pada penelitian ini adalah metode eksperimen, meliputi; (1) proses pengumpulan data, (2) pengolahan data, (3) perancangan sistem *online word processor*, 2FA, dan algoritma AES, (4) membangun sistem, dan (5) evaluasi sistem. Berikut adalah uraian dan penjelasan terkait tahap-tahap yang dilakukan pada saat melaksanakan proses penelitian:

3.1.1 Pengumpulan Data

Pada proses pengumpulan data, merupakan proses riset dengan membaca beberapa artikel di internet terkait permasalahan yang diangkat. Serta riset terkait beberapa jurnal terkait beberapa algoritma yang dapat dipakai untuk memecahkan permasalahan nantinya.

3.1.2 Pengolahan data

Pada proses pengolahan data, proses riset terkait kelemahan dan kelebihan masing-masing algoritma yang sudah dikumpulkan sebelumnya pada tahap pengumpulan data.

3.1.3 Perancangan sistem *online word processor*, 2FA, dan AES

Pada proses ini, merupakan proses perancangan alur dari aplikasi atau *website* yang akan dibangun yaitu *online word processor*. Serta perancangan mengimplementasikan 2FA dan enkripsi serta dekripsi menggunakan algoritma AES pada sistem yang dibangun. Perancangan dilakukan dengan membuat beberapa diagram alur atau *flowchart* serta rancangan tampilan dari sistem yang akan dibangun nantinya.

3.1.4 Pembangunan sistem

Proses ini merupakan proses implementasi 2FA dan algoritma AES terhadap *online word processor* yang dibangun. Pembangunan *website* dilakukan dengan

menggunakan bahasa pemrograman dan *framework* serta *library* yang telah ditentukan pada proses perancangan.

3.1.5 Evaluasi sistem

Proses ini merupakan proses melakukan percobaan *brute force attack* dan pengujian Avalanche Effect untuk memastikan hasil implementasi dilakukan dengan baik. Proses ini merupakan proses yang dilakukan secara rutin dalam beberapa saat untuk mencegah terjadinya *bug* atau *error* pada *website* yang sudah dibangun sebelumnya. Proses evaluasi dilakukan untuk meminimalisir celah pencurian atau pembobolan informasi milik *user*.

3.1.6 Penulisan Laporan

Proses ini merupakan proses akhir dengan menyelesaikan sebagian besar dari pembangunan sistem. Laporan yang berisi hasil keseluruhan dari penelitian yang dilakukan dengan terstruktur sesuai dengan kaidah penulisan laporan penelitian yang ada.

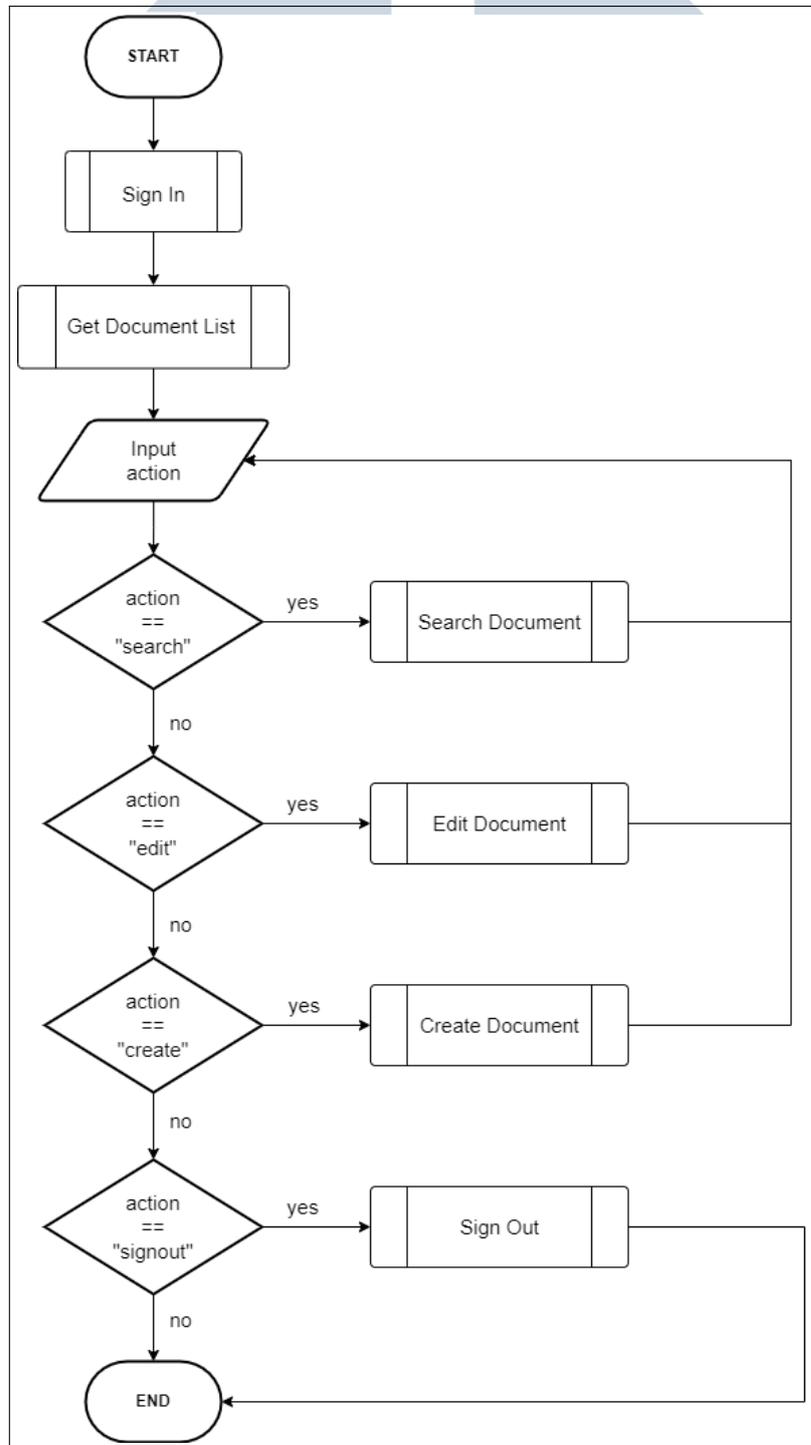
3.2 Perancangan sistem

Perancangan sistem ditujukan agar pada pembangunan sistem yaitu situs web *word processor* dapat memahami alur kerja yang dimiliki oleh sistem secara terstruktur. Perancangan sistem yang akan dijelaskan berupa *flowchart*, *database schema*, struktur basis data, dan *prototype* dari situs web.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A

3.2.1 Flowchart

A Flowchart - Home



Gambar 3.1. Flowchart - Home

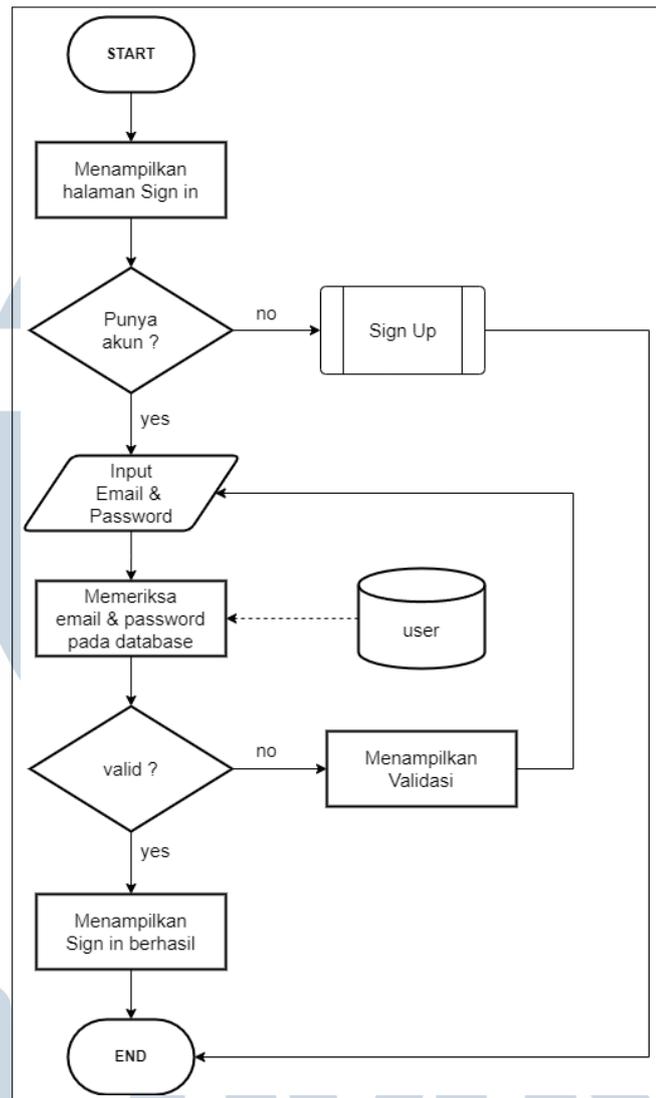
Pada gambar 3.1, merupakan alur dari halaman *home* dimana dapat dilihat bahwa saat *user* mengakses halaman *home* maka aplikasi akan meminta *user* untuk melakukan *sign in*. Sesaat setelah *user* melakukan *sign in*, maka sistem akan mengambil daftar dokumen yang sesuai dan ditampilkan pada halaman *home*. Pada halaman *home*, *user* dapat melakukan beberapa aksi atau interaksi dengan sistem yaitu *user* dapat melakukan pencarian dokumen berdasarkan judul. Selanjutnya, *user* dapat memilih dokumen yang ingin di *edit* pada daftar dokumen yang ada dan sistem secara otomatis akan mengarahkan *user* ke halaman *edit* dokumen.

Pada halaman *home*, terdapat tombol untuk menambahkan dokumen baru yang dapat ditekan oleh *user*. Pada saat *user* menekan tombol, maka sistem akan secara otomatis membuka modal *create new document*. Pada modal akan terdapat beberapa data wajib yang harus diisi oleh *user*. Saat data yang diberikan *user* valid maka sistem akan mengarahkan *user* ke halaman untuk melakukan perubahan pada isi dokumen. Terakhir, terdapat tombol *sign out* dimana apabila ditekan oleh *user* maka sistem akan menjalankan proses *sign out* terhadap *user* terkait.

B Flowchart - Sign In

Pada proses *sign in*, saat halaman diakses maka sistem akan menampilkan halaman *sign in*. Pada halaman ini, *user* diwajibkan untuk memiliki akun pribadi untuk dapat mengakses dokumen. Apabila *user* tidak memiliki akun maka sistem akan mengarahkan *user* ke halaman *sign up*. Sebaliknya, saat *user* sudah memiliki akun *user* akan diminta sistem untuk memasukkan *email* dan *password*. Hasil input dari *user* akan diperiksa oleh sistem apakah data valid atau tidak jika tidak valid sistem menampilkan validasi. Apabila data yang dimasukkan sesuai maka *user* akan diarahkan ke halaman *home*. Alur proses pada halaman *sign in* dapat dilihat pada gambar 3.2.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A

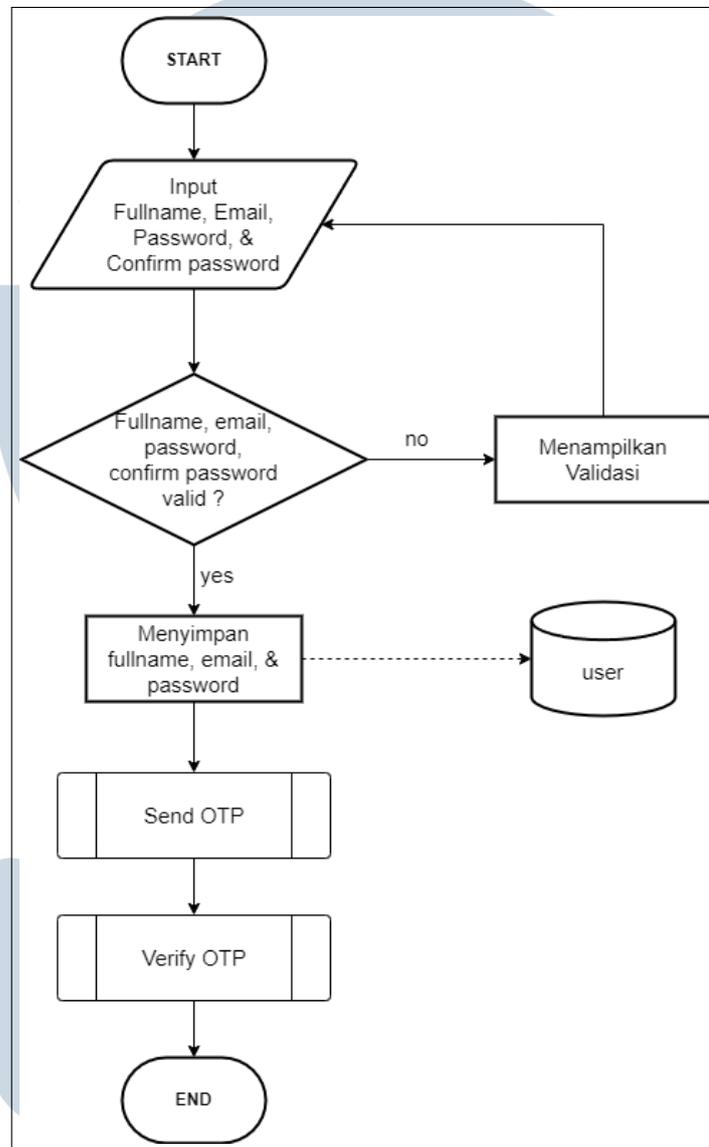


Gambar 3.2. Flowchart - Sign In

C Flowchart - Sign Up

Gambar 3.3 merupakan alur dari proses dari *sign up* atau pendaftaran akun. Sistem meminta *user* untuk memasukkan beberapa data yang diperlukan seperti nama lengkap, *email*, dan *password*. Data-data yang dimasukkan oleh *user* akan diperiksa apakah data yang dimasukkan valid atau tidak. Apabila data tidak valid maka sistem akan menampilkan validasi atau kesalahan yang dimasukkan oleh *user*. Sebaliknya, apabila data yang diberikan oleh *user* valid maka sistem akan menyimpan data yang dimasukkan ke dalam *database user*. Setelah data *user* selesai dibuat maka sistem akan mengirimkan OTP melalui *email* dan kemudian

akan melakukan verifikasi OTP sesuai kode yang dimasukkan oleh *user*. Setelah OTP berhasil terverifikasi maka *user* dapat menggunakan situs web.

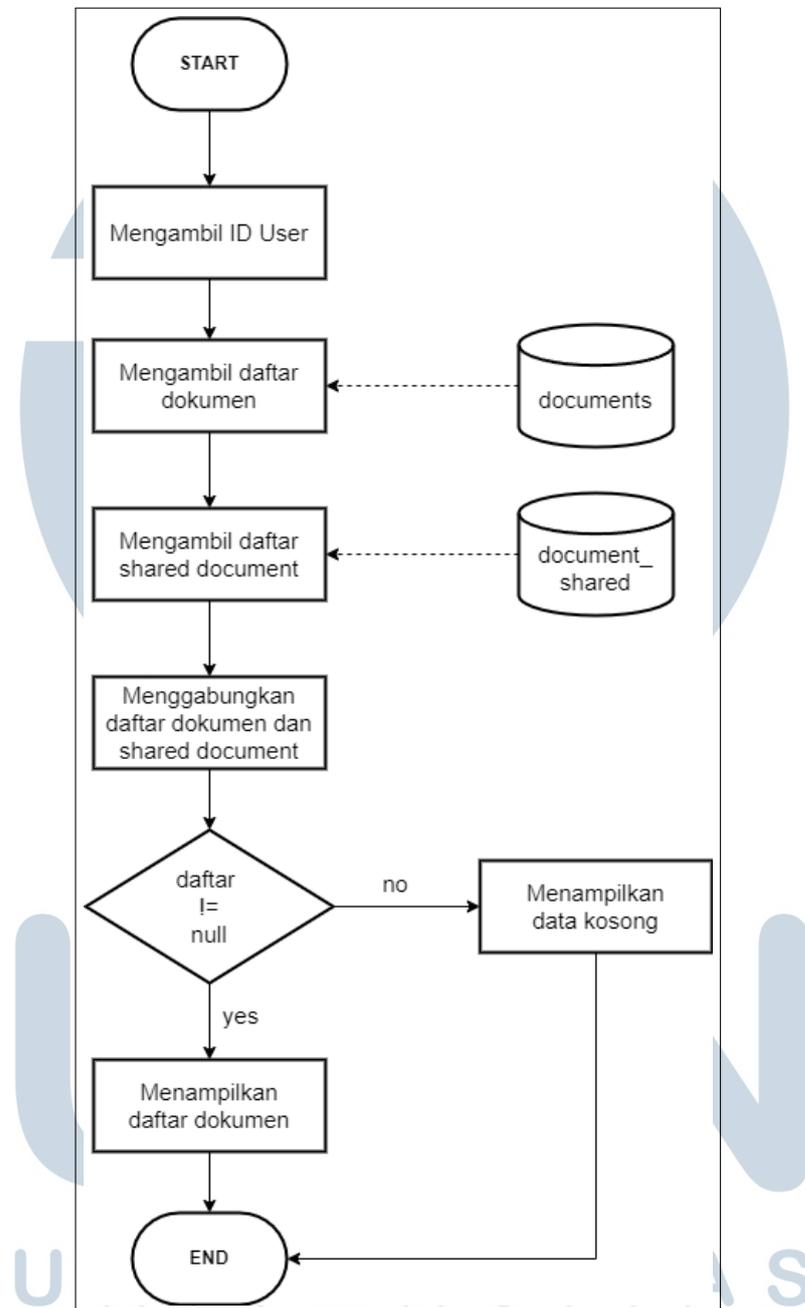


Gambar 3.3. Flowchart - Sign Up

D Flowchart - Get Document List

Gambar 3.4 merupakan gambaran alur dari proses mengambil daftar dokumen. Sistem sebelumnya akan mengambil id dari *user* dan dikirimkan ke *database*. Sistem akan mencari data yang sesuai pada *database* sesuai dengan id yang didapat. Sistem akan mengembalikan null atau kosong ketika tidak ada data yang sesuai pada *database*. Sistem akan mengembalikan daftar dokumen apabila

ada data yang tersedia.

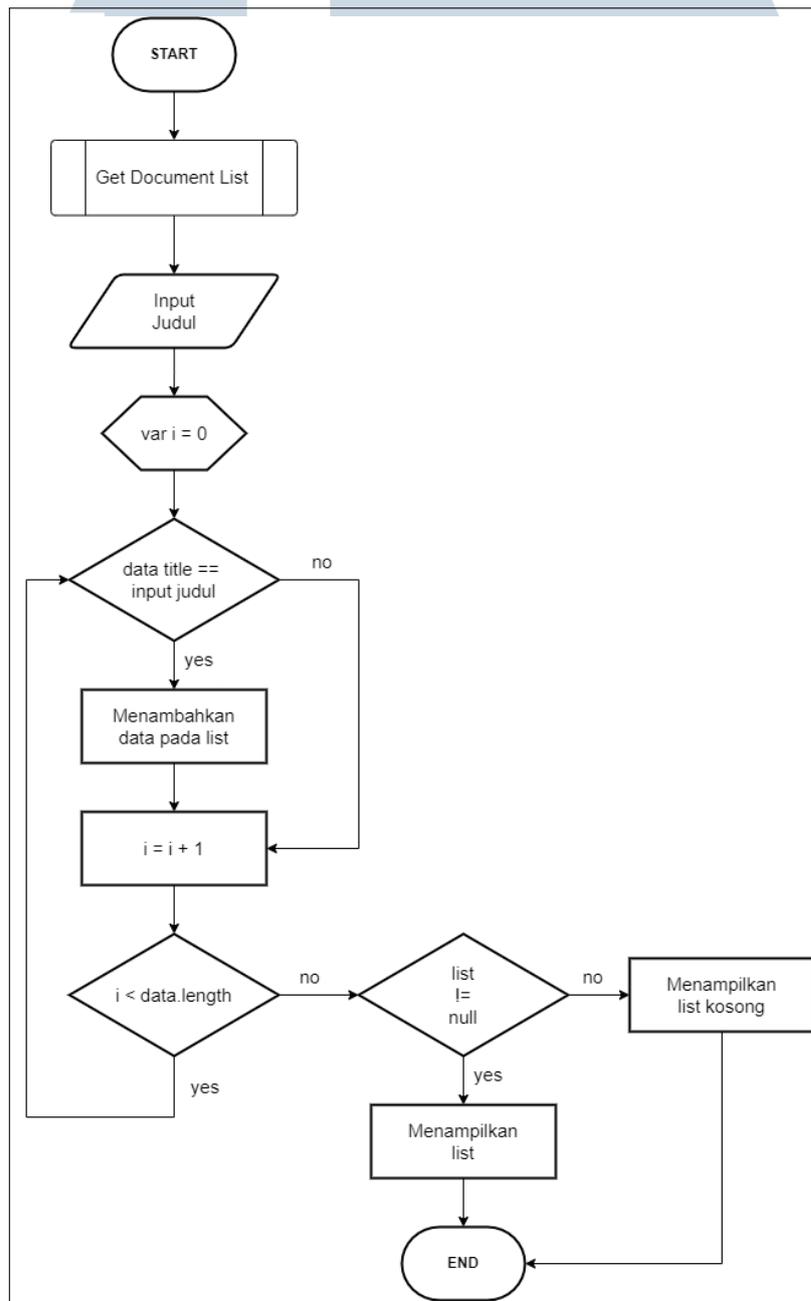


Gambar 3.4. Flowchart - Get Document List

E Flowchart - Search Document

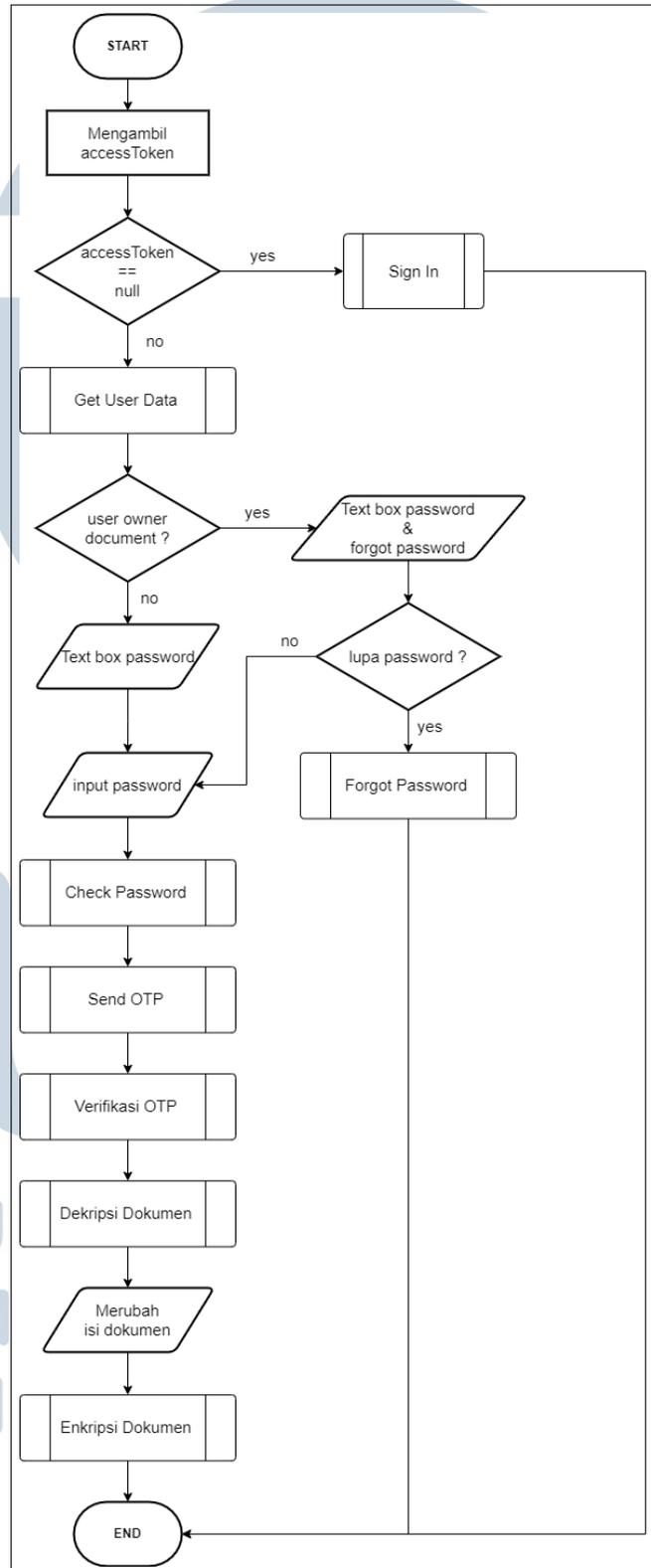
Gambar 3.5 memiliki keseluruhan fungsi yang dimiliki oleh *get document list* yang terdapat perbedaan pada satu kiriman data yaitu *search* judul. Hasil dari kiriman ini akan melakukan filter pada data yang didapatkan dari *database*. Filter

dilakukan pada bagian judul dari setiap dokumen yang didapatkan dan dicari yang sesuai dengan filter *search* judul yang dikirimkan. Sistem akan menambahkan data yang sesuai ke dalam *list* apabila tidak sesuai maka sistem tidak memasukkan ke dalam *list* dan hal ini diulangi ke semua data yang ada. Ketika *list* kosong maka sistem akan menampilkan kosong dan sebaliknya sistem akan menampilkan *list* yang sesuai.



Gambar 3.5. Flowchart - Search Document

F Flowchart - Edit Document



Gambar 3.6. Flowchart - Edit Document

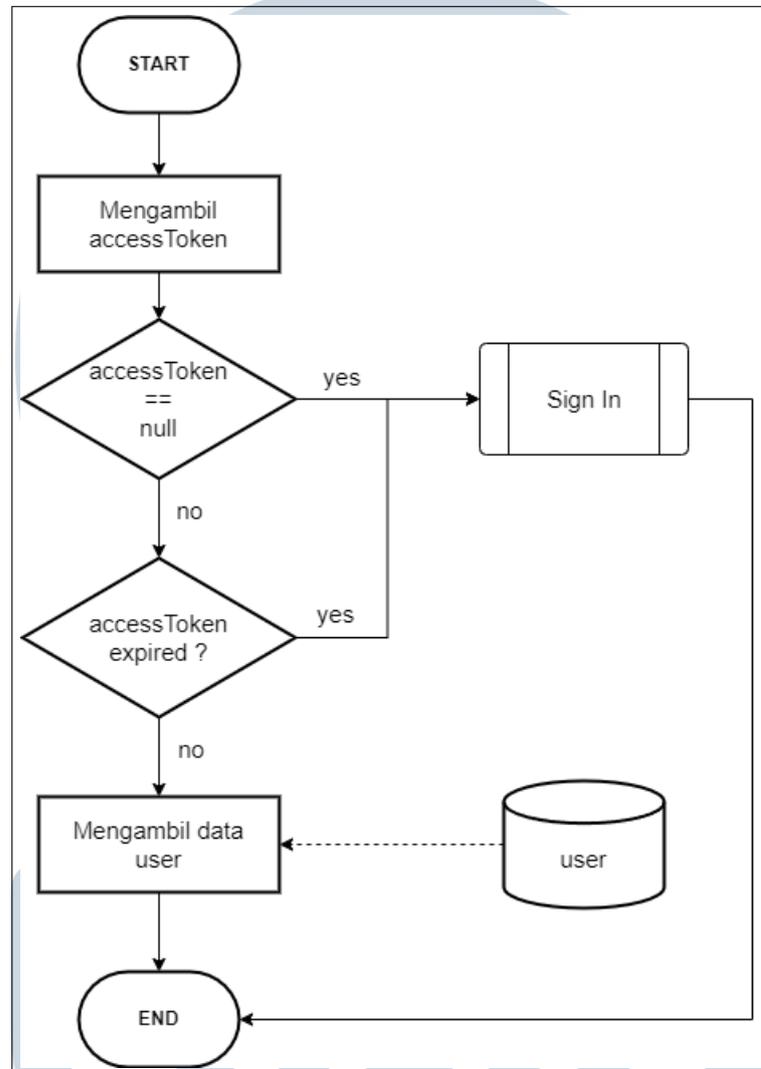
Pada gambar 3.6, menunjukkan keseluruhan alur dari proses *edit* dokumen. Sistem akan memeriksa akses token dari *user*, akses token bernilai null akan sistem arahkan ke halaman *sign in* sedangkan apabila token valid maka sistem akan mengambil data *user* yang sedang *sign in*. Token tersebut akan dikirimkan untuk mendapatkan data lengkap terkait *user* yang sedang *sign in*. Sistem juga akan mengambil detail dari dokumen yang idnya sesuai dengan pilihan *user* sebelumnya.

User yang didapatkan akan diverifikasi apakah *user* tersebut merupakan pemilik dari dokumen yang diakses atau tidak. Sistem akan menampilkan modal untuk meminta *user* untuk memasukkan *password* dari dokumen. Pada modal ini akan memiliki perbedaan apabila *user* adalah pemilik dokumen maka akan terdapat fitur tambahan pada modal yaitu *forgot password*. Fitur *forgot password* merupakan fitur khusus yang dibangun khusus untuk pemilik dokumen yang terdaftar sehingga *user* yang bukan pemilik dokumen tidak bisa mengganti *password* dari dokumen.

Sesaat setelah *user* mengakses fitur *forgot password* maka sistem akan mengarahkan *user* ke halaman *forgot password*. Setelah proses verifikasi selesai, maka sistem akan melanjutkan ke proses untuk melakukan pergantian *password* pada dokumen yang ingin diubah oleh pemiliknya. Akses dokumen akan diberikan kepada *user* setelah *user* berhasil mengganti *password* dokumennya.

Hal ini berbeda apabila *user* biasa atau saat pemilik tidak menekan *forgot password* yaitu sistem akan meminta *user* untuk memasukkan *password* dari dokumen yang ingin diakses. Input berupa *password* yang dikirimkan *user* akan diperiksa apakah *password* yang dimasukkan sesuai atau tidak. Apabila sesuai maka sistem akan mengirimkan OTP ke email *user* yang melakukan akses. Input OTP yang dimasukkan *user* akan diverifikasi dan bila data sesuai maka sistem akan melakukan dekripsi konten dari dokumen yang tersimpan pada *database*. Hasil dekripsi yang dilakukan oleh sistem akan ditampilkan isi dari dokumen tersebut sehingga dapat dilakukan perubahan oleh *user*. Perubahan yang dilakukan selanjutnya dienkripsi oleh sistem. Hasil dari enkripsi akan dikirimkan kembali ke dalam *database*.

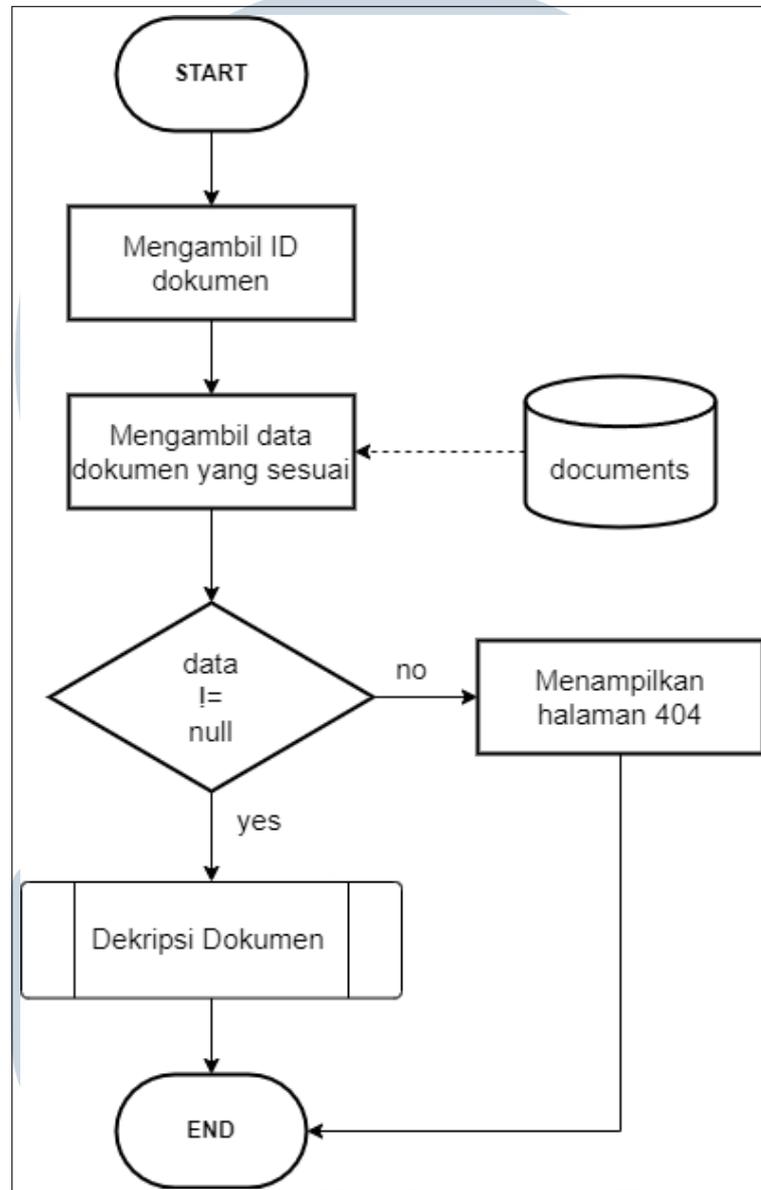
G Flowchart - Get User Data



Gambar 3.7. Flowchart - Get User Data

Gambar 3.7 merupakan alur dari proses mengambil data *user* yang sedang *sign in* ke dalam *database*. Data yang akan dikirimkan merupakan token apabila token bernilai kosong maka sistem akan mengarahkan *user* ke halaman *sign in*. Hal yang sama berlaku apabila token yang dikirimkan sudah *expired* atau melewati masa aktifnya. Apabila kedua kondisi sudah dipenuhi maka sistem akan mengambil data yang memiliki token tersebut dan mengambilnya dari *database user* yang terdaftar.

H Flowchart - Get Document Detail

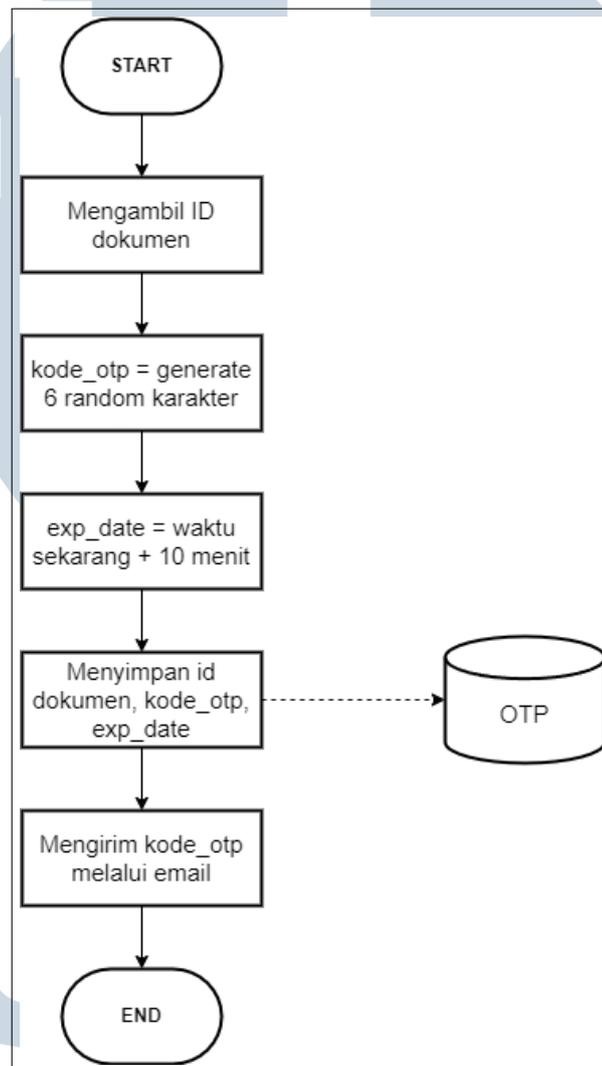


Gambar 3.8. Flowchart - Get Document Detail

Proses selanjutnya atau yang ditunjukkan pada gambar 3.8 adalah proses atau alur dari mengambil data dokumen secara lengkap dari *database*. Sistem akan mencari dokumen yang memiliki id yang sesuai dengan id dokumen yang ingin di akses pada tabel *document*. Apabila data tidak ditemukan maka sistem akan mengarahkan *user* ke halaman *not found*. Kemudian, sistem akan mengambil data detail dokumen dari table *document_details* pada *database* kemudian data seperti

dokumen *password* dan isi dari dokumen akan di dekripsi.

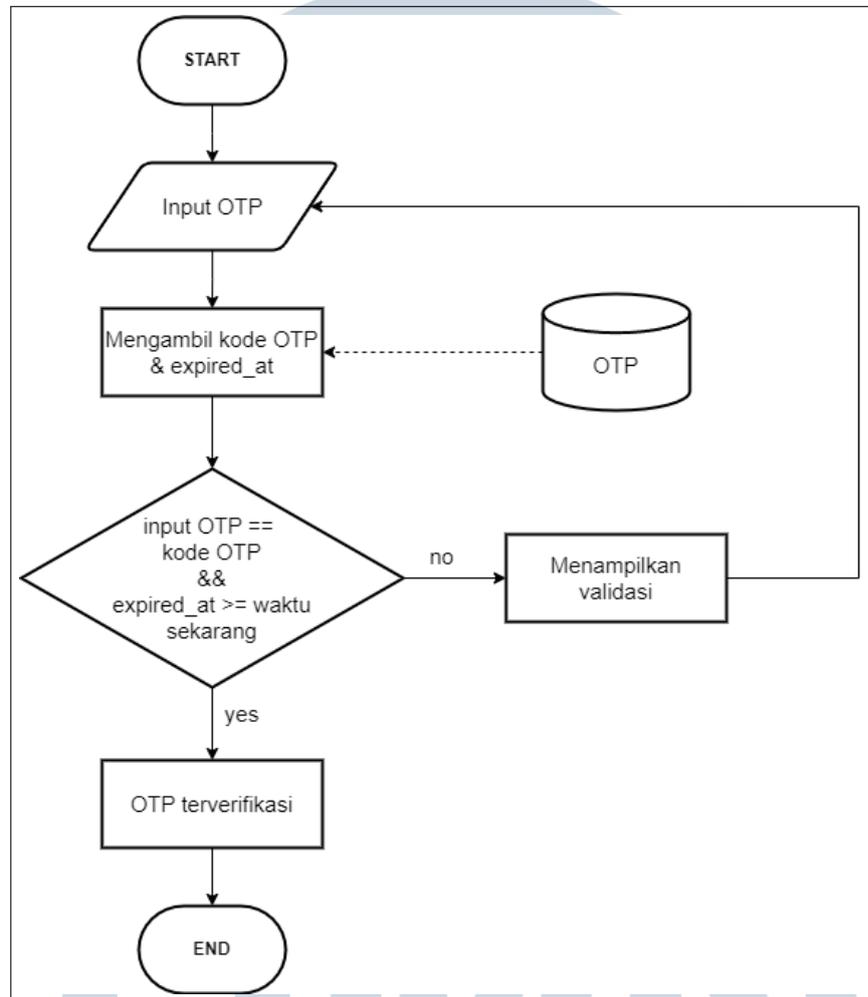
I Flowchart - Send OTP



Gambar 3.9. Flowchart - Send OTP

Gambar 3.9 merupakan alur dari proses mengirimkan OTP melalui *email*. Pada saat fitur diakses sistem akan mengambil id dokumen sesuai dengan dokumen yang dibuka. Sistem juga akan secara otomatis membuat enam karakter campuran yang mengandung angka dan huruf. Sistem kemudian akan menyimpan kode OTP tersebut ke dalam *database* dan menambahkan kolom *expired_at* yang berisikan waktu 10 menit saat OTP dibuat. Setelah itu sistem akan mengirimkan *email* kepada *user* yang berisikan kode OTP yang aktif hanya untuk 10 menit ke depan.

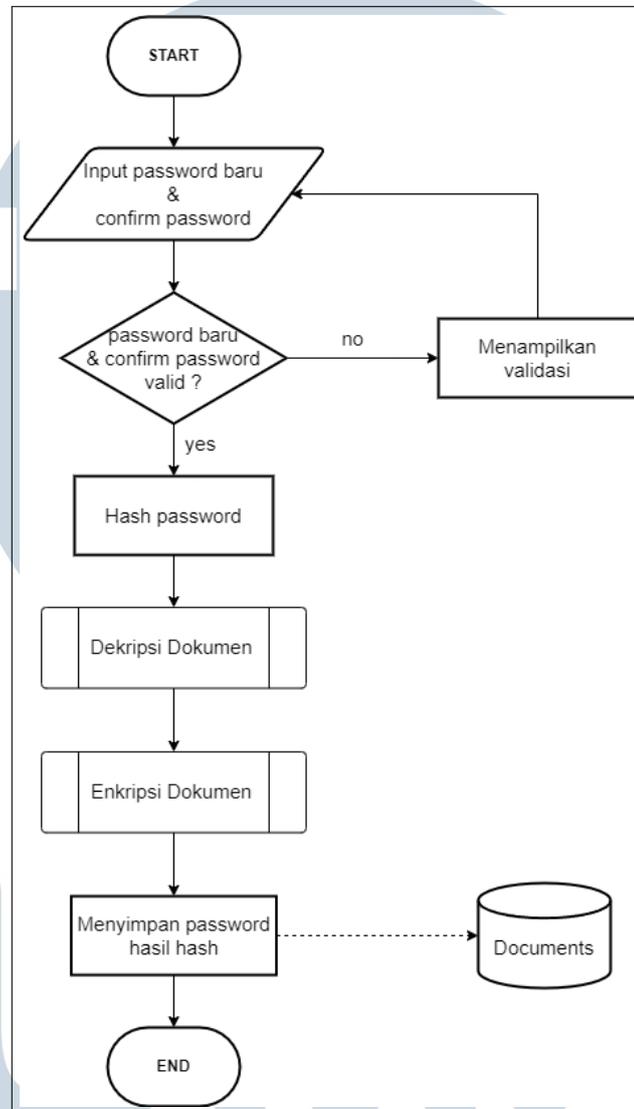
J Flowchart - Verifikasi OTP



Gambar 3.10. Flowchart - Verifikasi OTP

Gambar 3.10 merupakan alur proses dari verifikasi OTP. Sistem akan meminta *user* untuk memasukkan kode OTP yang didapatkan melalui via *email*. Sistem akan mencari data yang sesuai pada tabel OTP pada *database*. Ketika data tidak tersedia pada *database* maka sistem akan menampilkan validasi bahwa kode OTP tidak sesuai. Sistem juga akan mengirimkan validasi apabila data terdapat pada *database* namun waktu *expired* sudah lewat. Sistem akan mengembalikan status sukses apabila input yang diberikan sesuai dengan yang ada pada *database*.

K Flowchart - Forgot Password

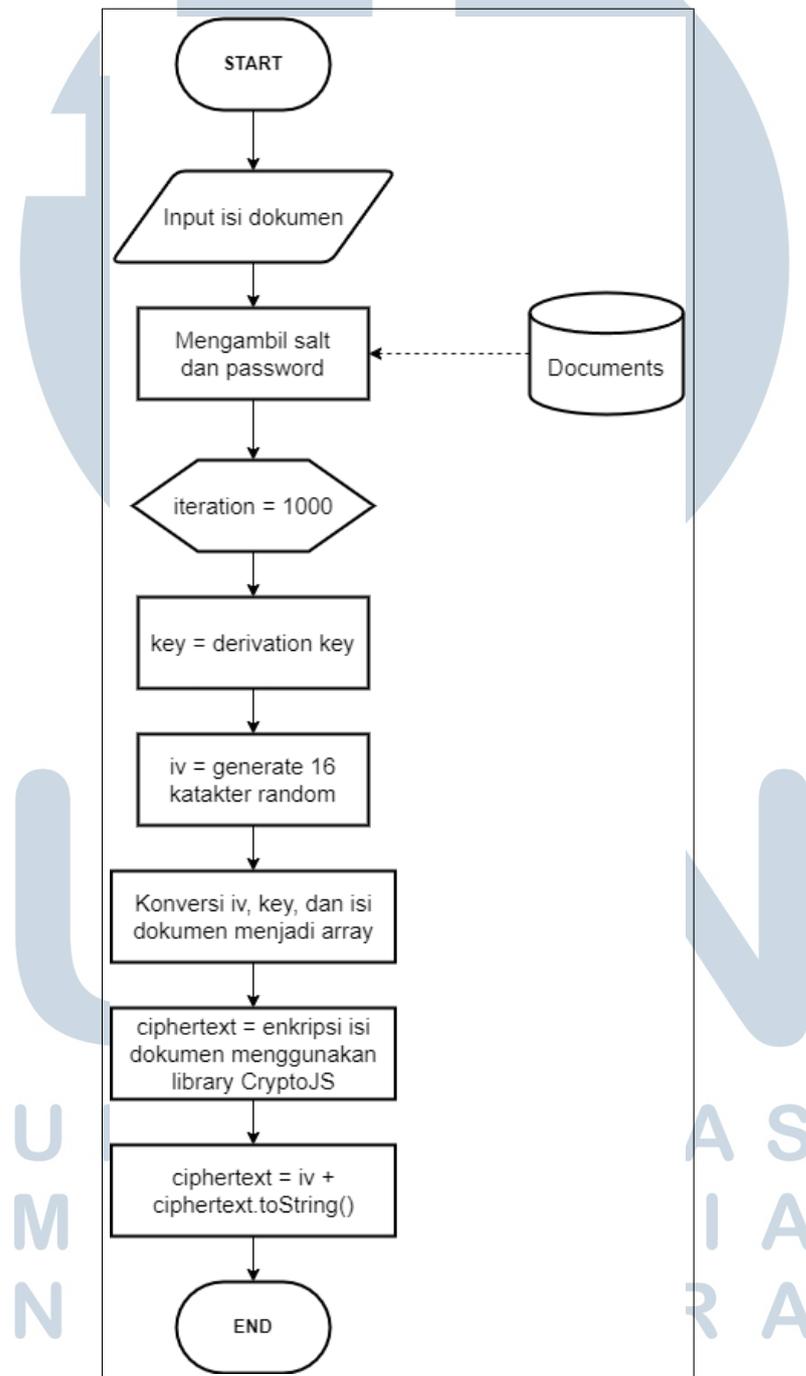


Gambar 3.11. Flowchart - Forgot Password

Gambar 3.11 menunjukkan alur proses dari *forgot password* atau *reset password*. Sistem akan meminta *user* untuk memasukkan *password* dan *confirm password* dan kemudian data tersebut diperiksa apakah jumlah karakter *password* sudah berjumlah delapan atau lebih serta *confirm password* dan *password* sudah sesuai. Apabila salah satu dari kedua kondisi tidak terpenuhi maka sistem akan menampilkan validasi *error*. *Password* yang dimasukkan akan dilakukan *hash* setelah itu sistem akan melakukan dekripsi pada isi dokumen dan kemudian dienkripsi kembali menggunakan *password* yang baru. Proses terakhir adalah

sistem menyimpan *password* hasil *hash* ke dalam *database*.

L Flowchart - Enkripsi Dokumen



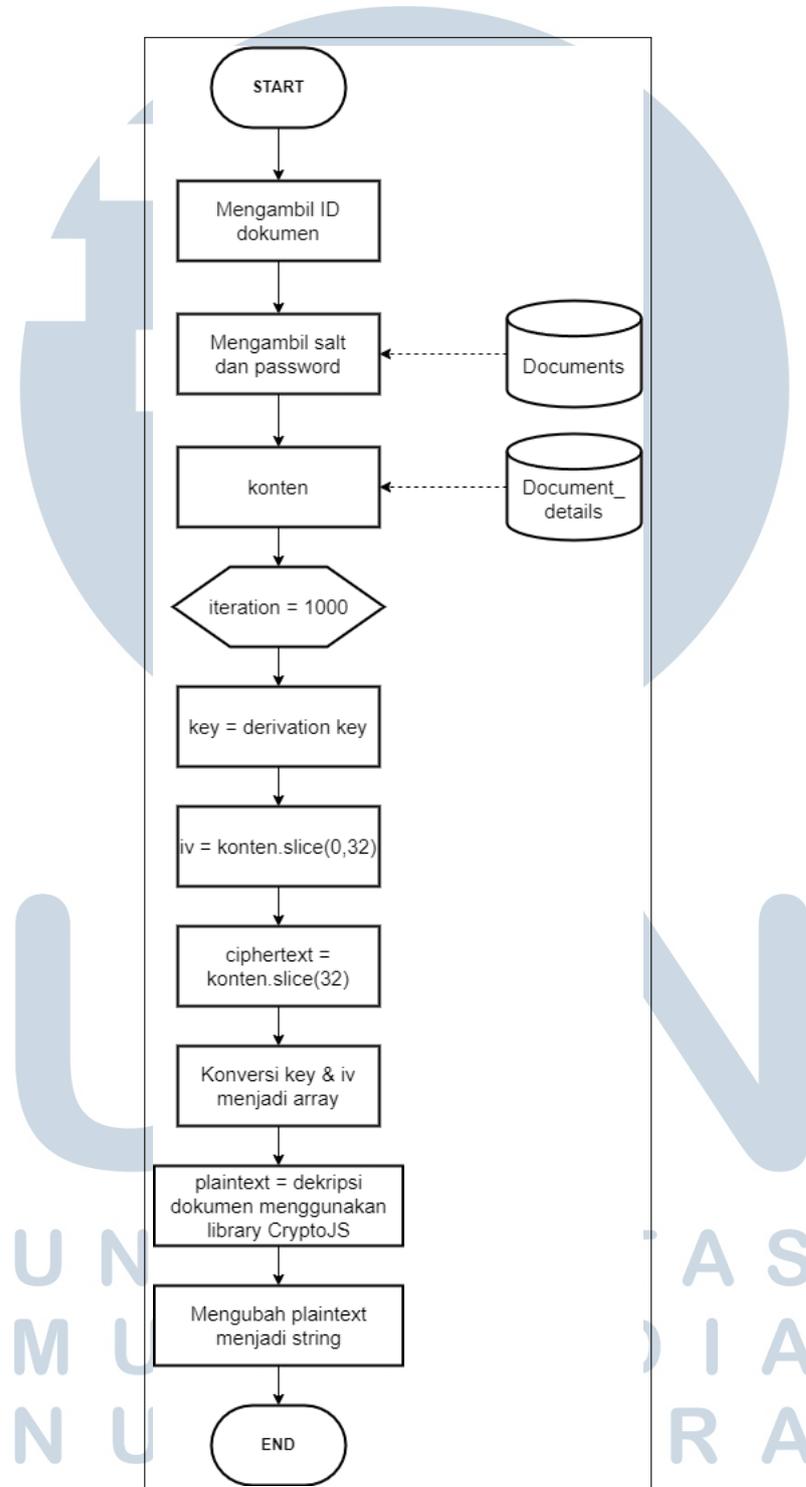
Gambar 3.12. Flowchart - Enkripsi Dokumen

Gambar 3.12 menunjukkan alur dalam proses enkripsi isi dari dokumen. Sistem akan mengambil *password* dan *salt* dari *database*. Sistem akan menentukan jumlah iterasi sebesar seribu kali iterasi. Sistem akan melakukan proses *derivation key* terhadap *password* menggunakan *salt* dan jumlah iterasi. Hasil yang didapatkan dari *derivation key* akan digunakan sebagai *key* untuk melakukan enkripsi. Serta sistem akan mengambil 16 karakter secara acak untuk dijadikan sebagai *initialization vector*.

Proses selanjutnya sistem akan melakukan konversi *key*, *initialization vector*, dan isi dokumen menjadi bentuk *array*. Setelah itu di enkripsi menggunakan AES-256 dengan bantuan CryptoJS. Hasil yang didapatkan digabungkan dengan nilai *initialization vector* dan diubah menjadi bentuk string agar mudah diproses ke proses selanjutnya yaitu disimpan atau *diupdate* ke dalam *database*.



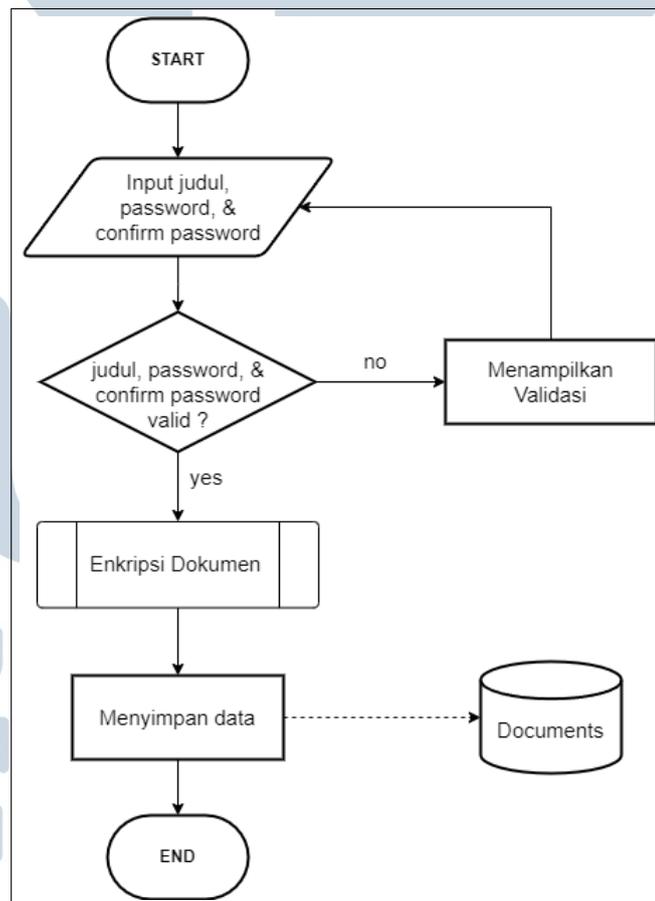
M Flowchart - Dekripsi Dokumen



Gambar 3.13. Flowchart - Dekripsi Dokumen

Gambar 3.13 merupakan alur dari perancangan proses dekripsi isi dari dokumen. Sistem akan mengambil data *password*, *salt*, dan konten dari dokumen pada *database*. *Password* dan *salt* yang didapatkan selanjutnya dilakukan proses *derivation key* untuk digunakan sebagai *key* untuk dekripsi. Sistem akan mengambil 32 karakter pertama pada konten dokumen yang terenkripsi untuk digunakan sebagai *initialization vector*. Pada sisa karakter yang ada sistem akan gunakan sebagai *cipher text* yang akan dilakukan proses dekripsi. Sistem akan melakukan konversi *key* dan *initialization vector* tersebut ke dalam bentuk *array*. Proses selanjutnya adalah proses dekripsi menggunakan *library* CryptoJS. Hasil dekripsi yang didapatkan akan diubah menjadi bentuk *string* sehingga dapat digunakan pada *word processor*.

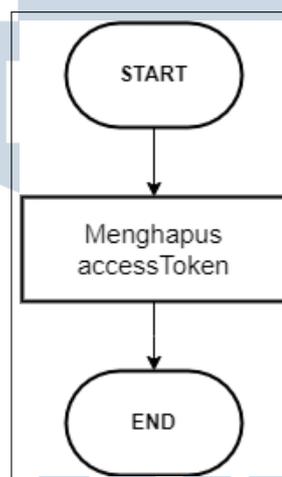
N Flowchart - Create Document



Gambar 3.14. Flowchart - Create Document

Gambar 3.14 merupakan alur dari proses pembuatan dokumen baru. Dimana sistem akan secara otomatis menampilkan modal yang meminta *user* untuk memasukkan judul dari dokumen serta *password* yang akan digunakan untuk menjaga keamanan dokumen. Data-data yang diinput oleh *user* akan diperiksa, ketika titlenya bernilai null atau *password* kurang dari 8 karakter serta *confirm password* tidak sama dengan *password* maka sistem akan mengirimkan validasi sesuai dengan kondisi masing-masing. Jika semua kondisi terpenuhi maka sistem akan melakukan proses enkripsi *password* yang diinput oleh user kemudian dokumen kosong akan di enkripsi juga agar dapat disimpan ke dalam *database* dengan mudah.

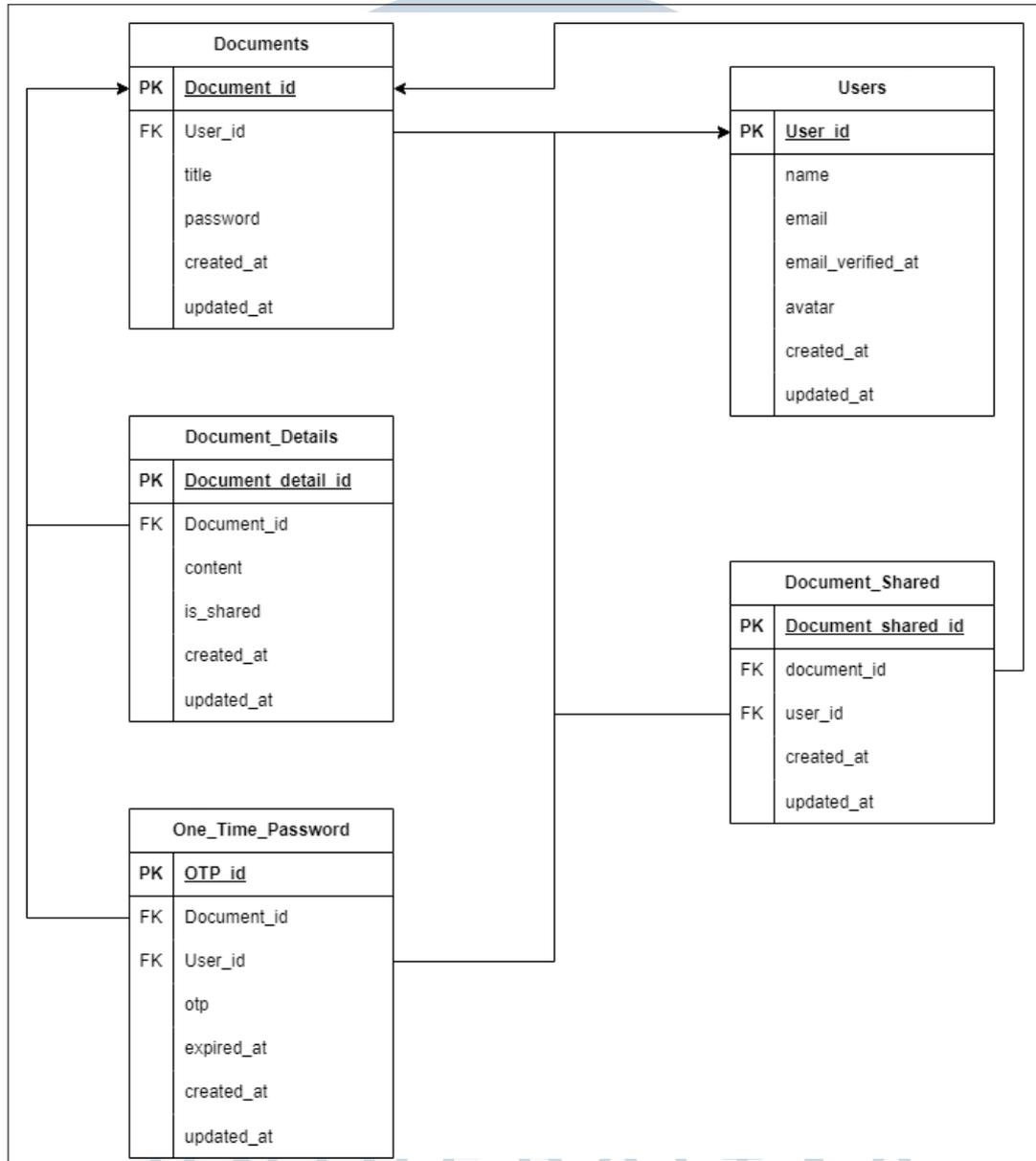
O *Flowchart - Sign Out*



Gambar 3.15. *Flowchart - Sign out*

Gambar 3.15 merupakan alur dari proses *sign out*, proses dari alur ini singkat yaitu sistem akan menghapus *access token* yang digunakan oleh *user* yang ingin keluar. Dari proses ini membuat *user* akan kehilangan akses untuk mengakses dokumen maupun aplikasi secara keseluruhan.

3.2.2 Database Schema



Gambar 3.16. Database Schema

Gambar 3.16 merupakan gambar dari skema basis data atau *database schema* dari *database* yang akan dibangun. Dimana terdapat lima buah tabel *database* yang akan digunakan pada sistem. Tabel pertama merupakan table *user* yang akan menyimpan data-data dari *user* yang melakukan *sign in* atau *register*. Tabel Kedua merupakan tabel *documents* dimana tabel ini digunakan untuk menyimpan data-data dari inti dari *document* serta tabel ini memiliki hubungan

dengan tabel user. Hubungan yang terdapat antara kedua tabel merupakan terkait *user* yang memiliki kepemilikan dari dokumen.

Tabel *document details* yang digunakan untuk menyimpan data-data terkait dokumen yang lebih lengkap yaitu seperti isi dari dokumen dan tabel ini memiliki hubungan dengan tabel dokumen sebagai detail dari dokumen yang terhubung. Tabel selanjutnya adalah tabel *document shared* adalah tabel yang berisi informasi terkait *user* yang pernah mengakses masing-masing dokumen dan tabel ini terhubung dengan tabel *document* dan *user*. Terakhir, tabel *one time password* yang dipergunakan untuk menyimpan data sementara untuk verifikasi identitas *user*. Serta tabel memiliki hubungan terhadap *user* dan *document*.

3.2.3 Struktur Tabel Basis Data

A Tabel users

Tabel 3.1. Tabel *users*

Nama Kolom	Tipe Data	Panjang
user_id (PK)	Char	36
name	Varchar	36
email	Varchar	50
password	Varchar	255
email_verified_at	Timestamp	
avatar	Varchar	255
created_at	Timestamp	
updated_at	Timestamp	

Tabel 3.1 merupakan struktur tabel dari tabel *user* yang bertujuan untuk menyimpan data-data milik *user* yang melakukan *sign up*. Dimana data-data yang dimasukkan oleh *user* akan disimpan ke dalam tabel *user* setelah memenuhi semua syarat yang ada. Pada tabel ini memiliki *primary key* yang memiliki tipe data *char* karena menyimpan *uuid*. Pada proses *sign in*, sistem juga akan menggunakan tabel *user* untuk memastikan *email* dan *password* yang dimasukkan oleh *user*.

B Tabel *documents*

Tabel 3.2. Tabel *documents*

Nama Kolom	Tipe Data	Panjang
document_id (PK)	Char	32
user_id (FK)	Char	36
title	Varchar	120
password	Varchar	255
salt	Varchar	255
created_at	Timestamp	
updated_at	Timestamp	

Tabel 3.2 merupakan struktur tabel dari tabel *documents* dimana bertujuan untuk menyimpan data utama dari dokumen. Pada tabel ini terdapat *primary key* yaitu *document_id* yang akan menyimpan data berupa uuid. Serta tabel ini terhubung dengan tabel *user* melalui *user_id*.

C Tabel *document_details*

Tabel 3.3. Tabel *document_details*

Nama Kolom	Tipe Data	Panjang
document_details_id (PK)	Char	36
document_id (FK)	Char	36
content	Text	
is_shared	TinyInt	1
created_at	Timestamp	
updated_at	Timestamp	

Tabel 3.3 merupakan struktur tabel dari *document_details* yang akan menyimpan data pelengkap milik masing-masing dokumen salah satunya adalah konten atau isi dari dokumen. Tabel ini memiliki *primary key* dan *foreign key* berbentuk uuid yang dimana tabel ini akan terhubung dengan *parentnya* yaitu tabel *documents*.

D Tabel *document_shared*

Tabel 3.4. Tabel *document_shared*

Nama Kolom	Tipe Data	Panjang
document_shared_id (PK)	Char	36
document_id (FK)	Char	36
user_id (FK)	Char	36
created_at	Timestamp	
updated_at	Timestamp	

Table 3.4 merupakan struktur tabel dari tabel *document_shared* yang berfungsi untuk menyimpan *user* yang pernah mengakses dokumen milik orang lain. Tabel ini memiliki *primary key* berbentuk *uuid* dan memiliki dua buah *foreign key* yang masing-masing terhubung dengan tabel *documents* dan tabel *user*.

E Tabel *one_time_password*

Tabel 3.5. Tabel *one_time_password*

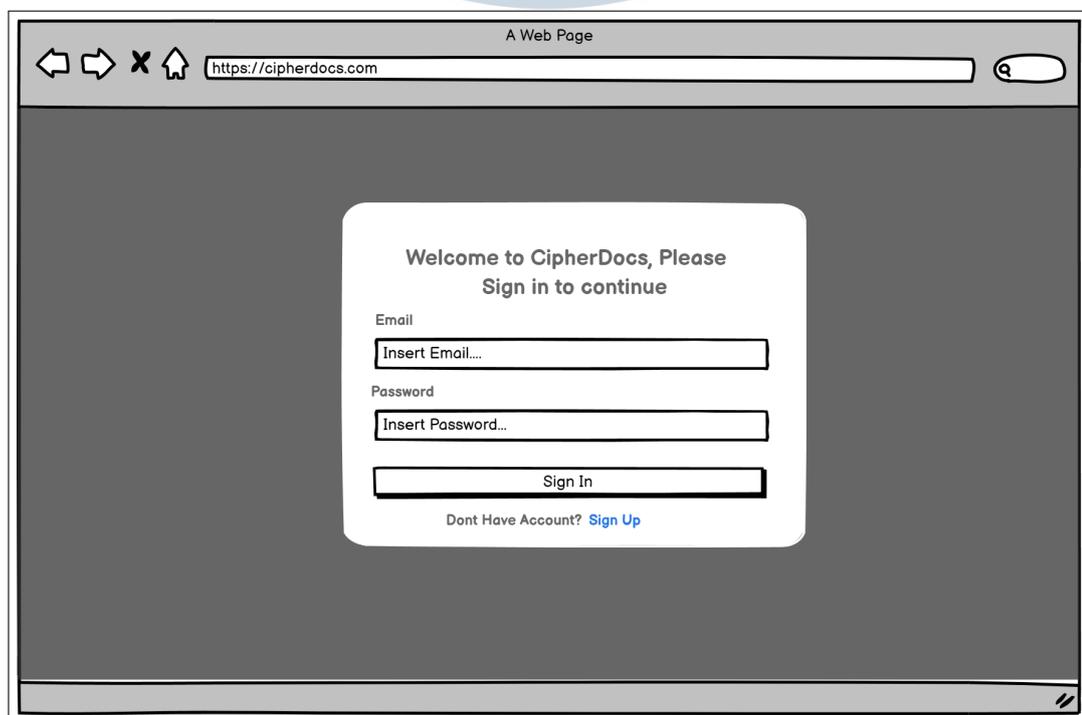
Nama Kolom	Tipe Data	Panjang
otp_id (PK)	Char	36
document_id (FK)	Char	36
user_id (FK)	Char	36
otp	Char	6
expired_at	Timestamp	
created_at	Timestamp	
updated_at	Timestamp	

Tabel 3.5 merupakan struktur tabel dari tabel *one_time_password* pada *database* yang berfungsi untuk menyimpan kode *one time password* (OTP). Data-data yang ada pada tabel ini hanya bersifat sementara dan memiliki hubungan dengan tabel *documents* dan juga tabel *user*.

3.2.4 *Prototype* Tampilan Situs Web

A Tampilan *Sign In*

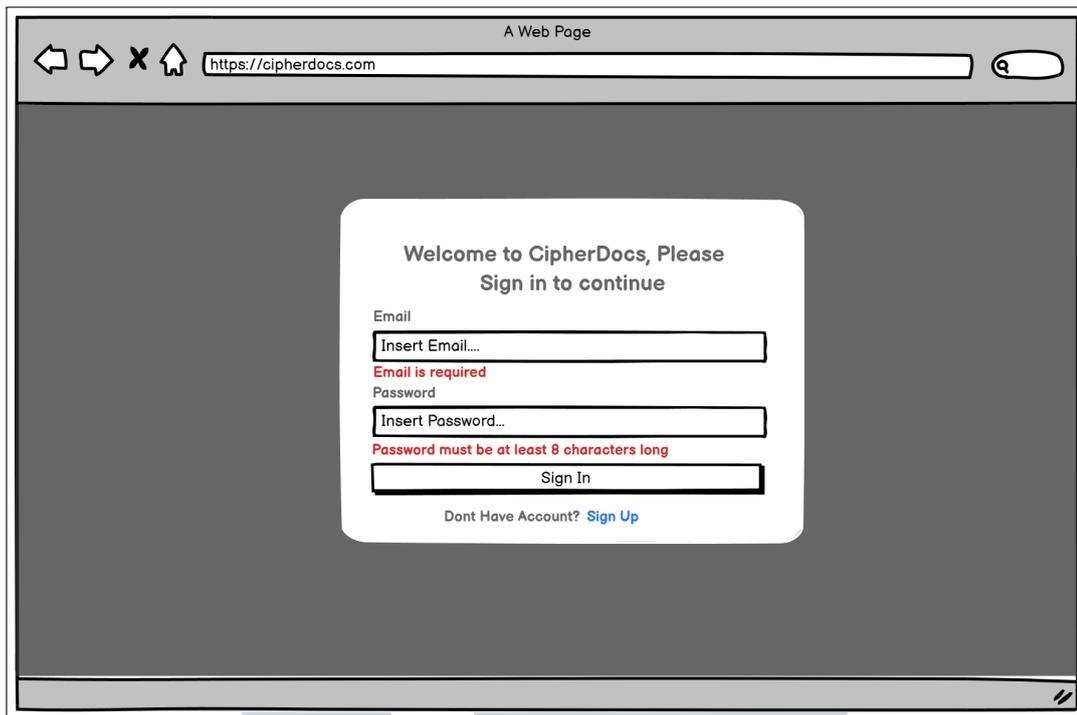
Gambar 3.17 merupakan rancangan tampilan antarmuka halaman *sign in* pada situs web. Gambar 3.18 merupakan rancangan tampilan antarmuka validasi pada halaman *sign in* ketika *user* menekan tombol tanpa mengisi data apapun. Pada halaman *sign in* terdapat *header* atau teks yang meminta *user* untuk melakukan *sign in* terlebih dahulu sebelum dapat mengakses dokumen. Dimana terdapat dua buah *text box* yaitu untuk *email* dan *password*. Pada halaman terdapat tombol *sign in* untuk sistem dapat memeriksa apakah data yang dimasukkan valid atau tidak. Apabila valid maka sistem akan mengarahkan ke halaman *home* sedangkan apabila tidak valid maka sistem akan menampilkan *error message*. Validasi yang akan ditampilkan apabila *email* kosong maka sistem akan menampilkan *email is required* atau *email* wajib diisi dan *password must be at least 8 characters long* atau *password* harus berjumlah 8 karakter atau lebih. Serta pada halaman terdapat satu *hyperlink* yang akan mengarahkan ke halaman *sign up* apabila *user* tidak memiliki akun.



The image shows a web browser window with the address bar displaying "https://cipherdocs.com". The main content area features a white sign-in form centered on a dark gray background. The form contains the following elements:

- Header text: "Welcome to CipherDocs, Please Sign in to continue"
- Email field: A text input box with the placeholder "Insert Email..."
- Password field: A text input box with the placeholder "Insert Password..."
- Sign In button: A rectangular button with the text "Sign In"
- Sign Up link: A blue hyperlink with the text "Dont Have Account? Sign Up"

Gambar 3.17. Tampilan *Sign In*

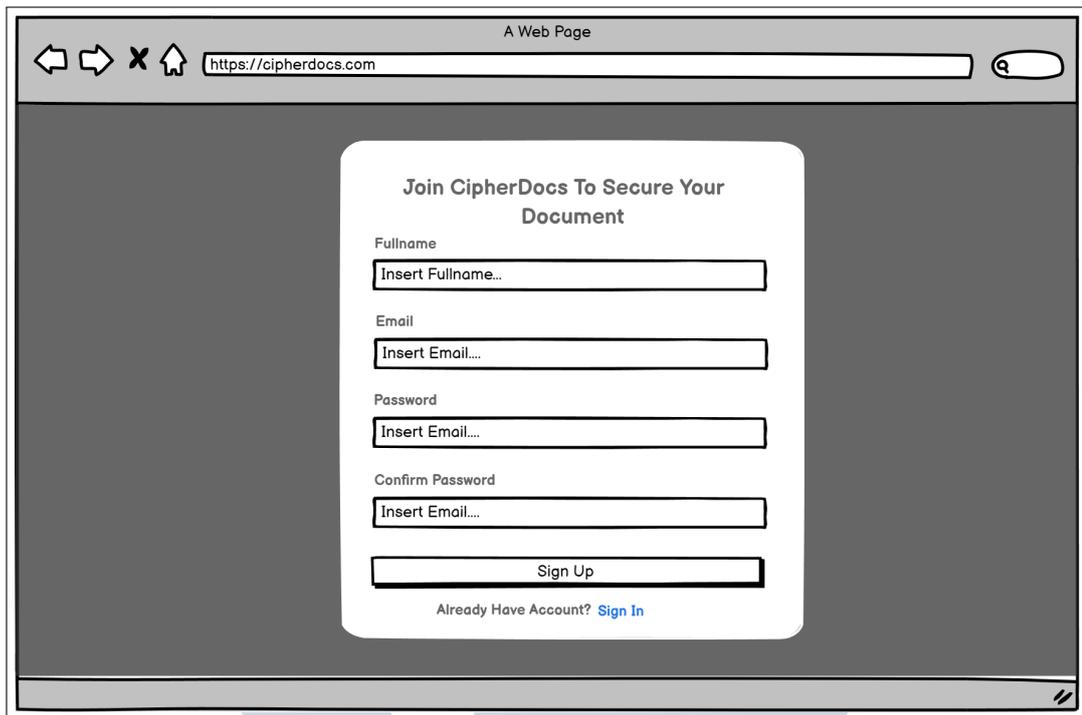


Gambar 3.18. Tampilan *Sign In* - Validasi Kosong

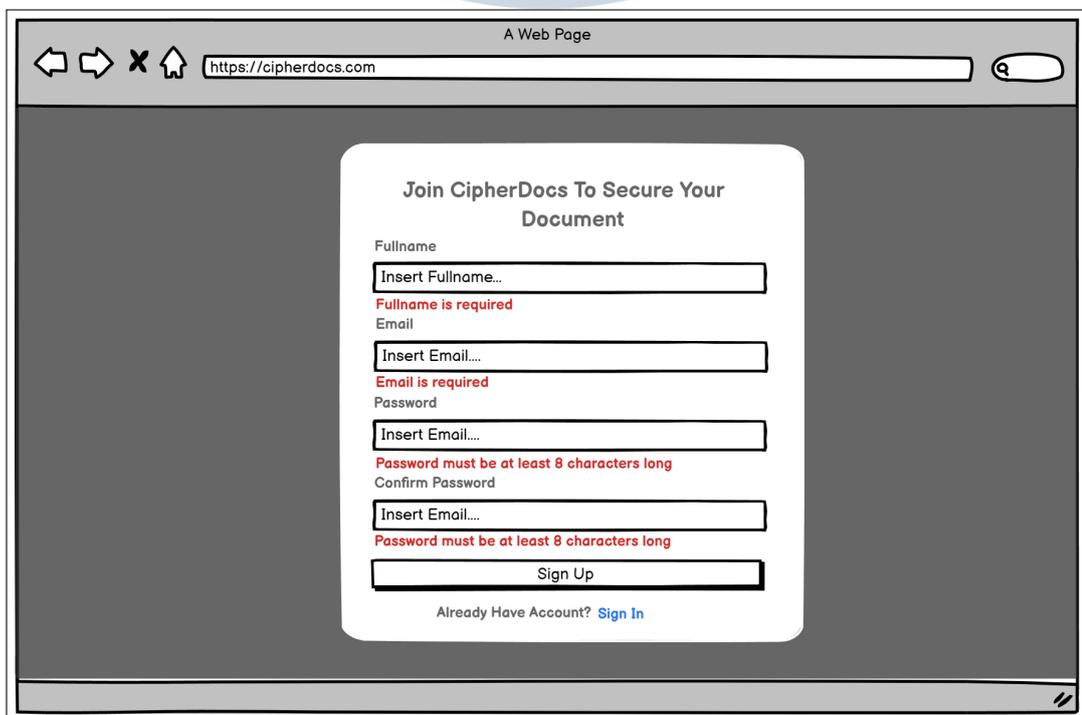
B Tampilan *Sign Up*

Gambar 3.19 merupakan rancangan tampilan antarmuka pada halaman *sign up* atau tampilan pendaftaran. Gambar 3.20 merupakan rancangan tampilan antarmuka validasi pada saat *user* menekan tombol *sign up* dengan keadaan kosong. Pada halaman *sign up* terdapat beberapa *text box* berupa *full name* atau nama lengkap, *email*, *password*, dan *confirm password*. Pada halaman *sign up* juga terdapat satu buah tombol berupa tombol *sign up*. Dimana tombol ini berfungsi untuk sistem memeriksa apakah data yang dimasukkan *user* valid atau tidak.

Sejalan dengan penjelasan sebelumnya, apabila tidak valid maka sistem akan menampilkan validasi dan apabila valid maka sistem akan menyimpan data *user* ke dalam *database*. Validasi yang ada pada halaman ketika *user* memasukkan data kosong maka sistem akan menampilkan validasi *fullname is required* pada kolom *fullname*, *email is required* pada kolom *email*, *password must be at least 8 characters long* pada kolom *password* dan *confirm password*. Serta terdapat *hyperlink* yang akan mengarahkan *user* ke halaman *sign in*.



Gambar 3.19. Tampilan *Sign Up*



Gambar 3.20. Tampilan *Sign Up* - Validasi Kosong

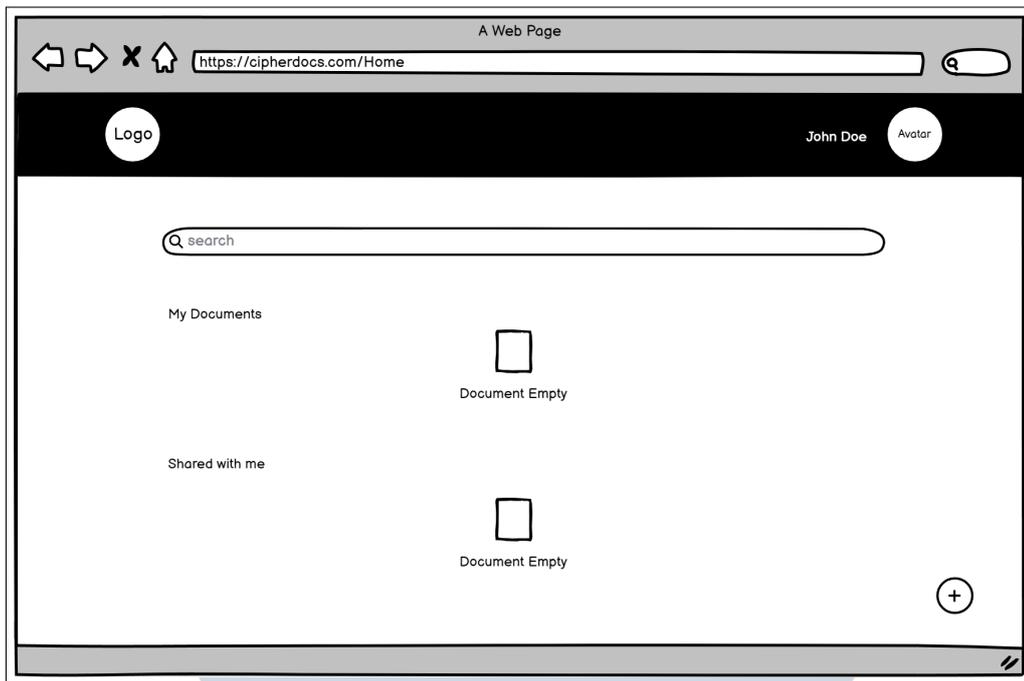
C Tampilan *Home*

Gambar 3.21 merupakan rancangan dari tampilan pada halaman *home* ketika *user* melakukan *sign in* untuk pertama kalinya atau tidak memiliki dokumen apapun. Gambar 3.22 merupakan rancangan antarmuka dari tampilan halaman *home* ketika *user* memiliki dokumen dan pernah melakukan akses terhadap dokumen *user* lain. Gambar 3.23 adalah rancangan antarmuka dari tampilan *home* dalam kondisi *user* melakukan *search* atau pencarian dokumen berdasarkan judul dari dokumen.

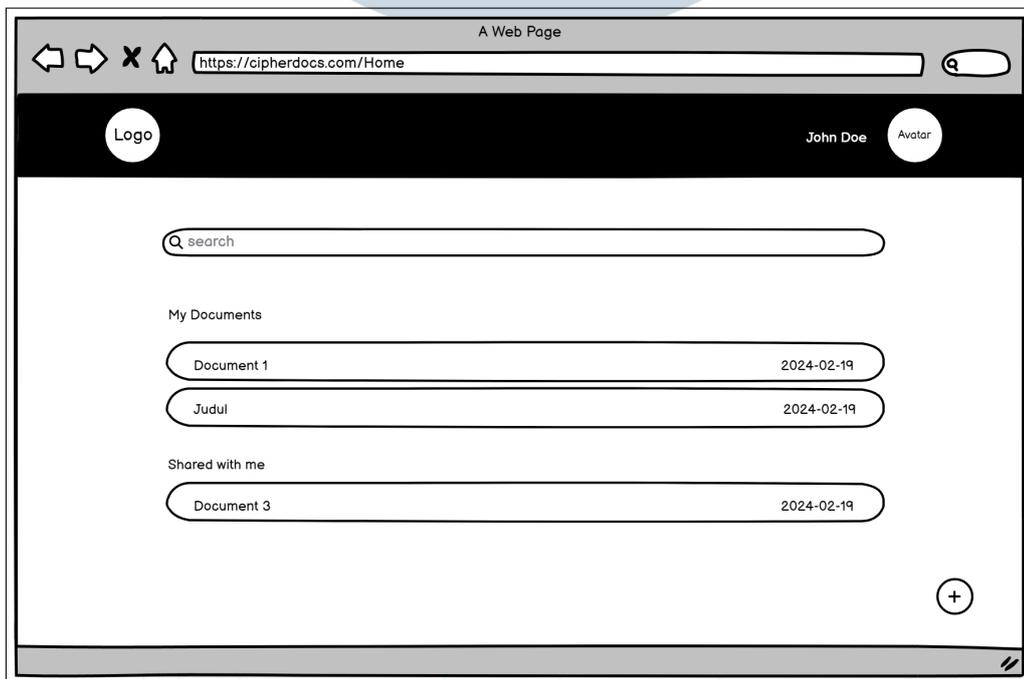
Pada halaman *home* dapat dilihat pada gambar terdapat *navigation bar* yang berisi logo serta informasi *user* yang sedang *sign in*. Pada bagian bawah *navigation bar* terdapat satu buah *user input* yaitu *search box* yang berfungsi untuk melakukan filter dari dokumen berdasarkan judulnya. Serta terdapat dua buah sesi untuk memisahkan dokumen milik *user* yaitu *my documents* dan *shared with me* yang merupakan sesi dokumen dari dokumen milik *user* lain yang pernah diakses oleh *user*.

Pada halaman ini, sistem akan menampilkan gambar dan tulisan informasi *document empty* ketika *user* tidak memiliki dokumen apapun baik milik sendiri maupun milik *user* yang lain. Pada saat *user* memiliki dokumen maka tampilan informasi tersebut tidak akan tampil melainkan sistem akan menampilkan daftar dokumen yang memiliki informasi judul dari dokumen serta waktu terakhir dokumen diupdate. Terakhir akan terdapat *floating button* yang berfungsi untuk mengarahkan *user* untuk membuat dokumen baru.

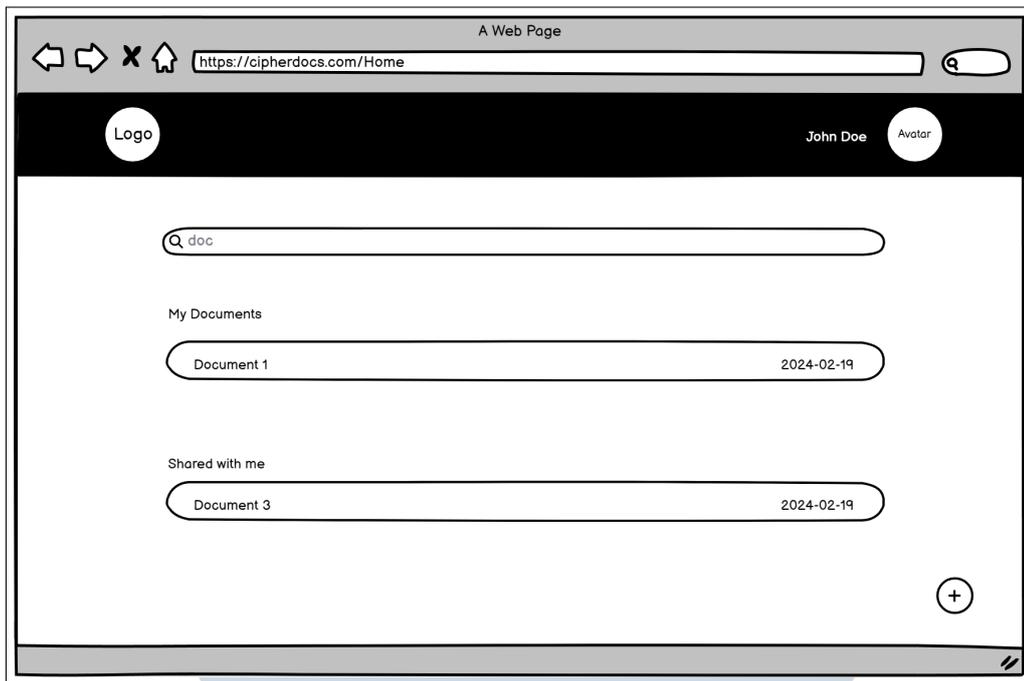
U M N
U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A



Gambar 3.21. Tampilan *Home* tanpa *document*



Gambar 3.22. Tampilan *Home*

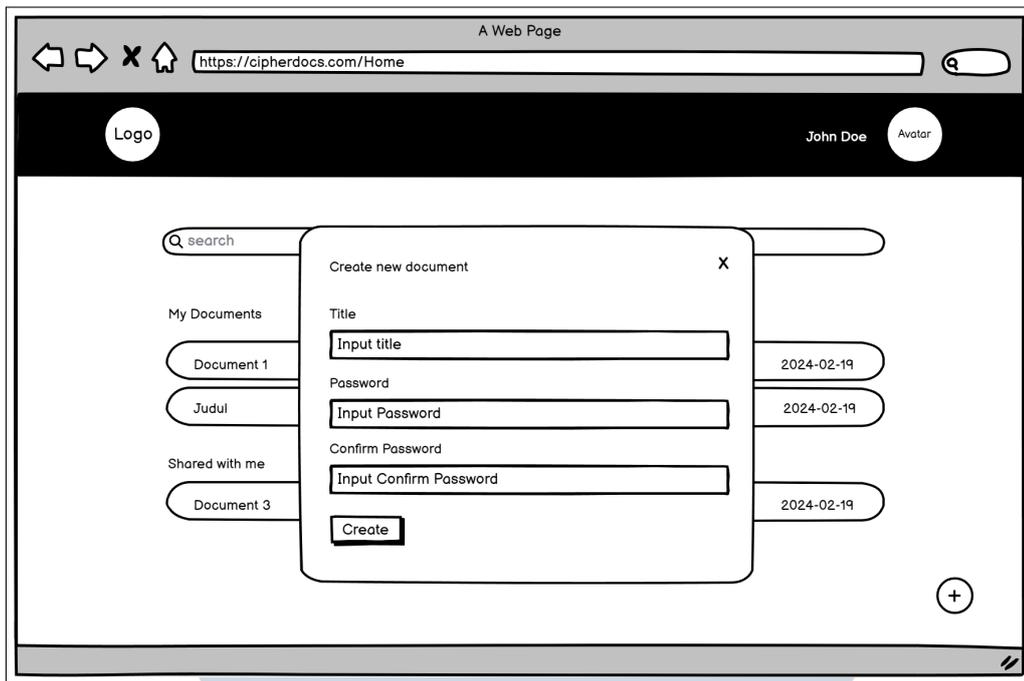


Gambar 3.23. Tampilan *Search Home*

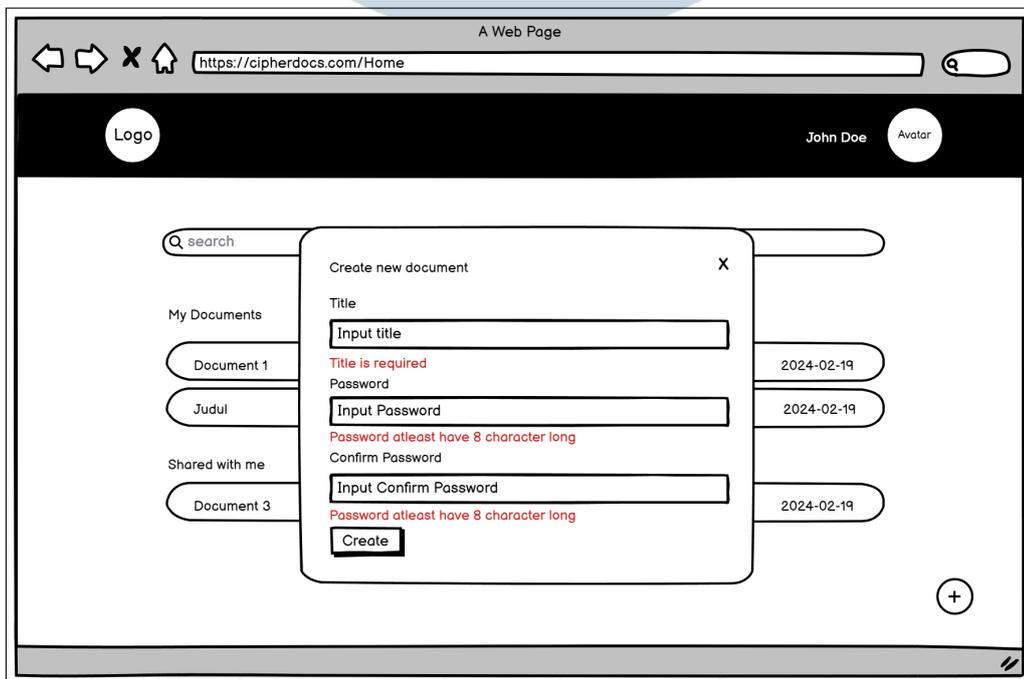
D Tampilan *Create Document*

Rancangan tampilan antarmuka ini merupakan rancangan tampilan setelah *user* menekan tombol *floating button* pada tampilan *home*. Gambar 3.24 merupakan rancangan tampilan dari membuat dokumen baru. Gambar 3.25 merupakan rancangan tampilan antarmuka pada saat data yang diinput *user* kosong. Gambar 3.26 merupakan rancangan tampilan antarmuka pada saat *password* dan *confirm password* tidak sama atau tidak sesuai.

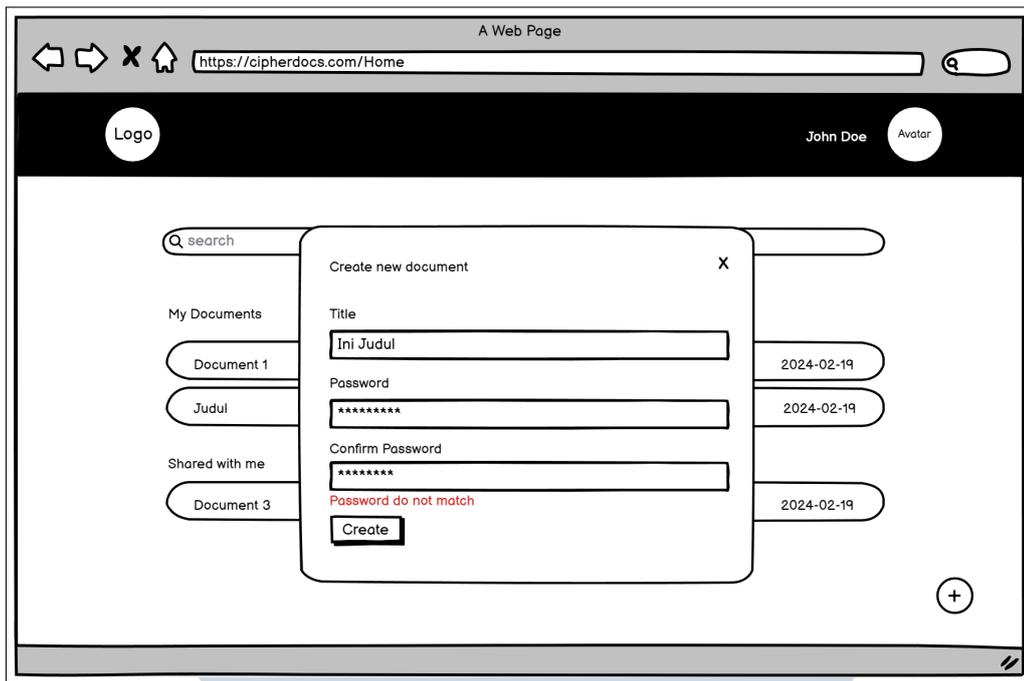
Pada tampilan ini rancangan merupakan sebuah modal atau *pop up* pada situs web yang dimana terdapat *title* yaitu *create new document* yang melambangkan modal untuk membuat dokumen baru. Terdapat tiga buah *text box* atau *input box* yaitu untuk *title* atau judul dokumen yang ingin dibuat, *password* untuk digunakan mengamankan dokumen, dan *confirm password* yaitu mencegah *user* melakukan kesalahan saat membuat *password*. Serta terdapat satu buah tombol untuk mengarahkan *user* agar dapat melakukan perubahan isi dari dokumen yang ada. Terdapat berbagai validasi ketika *user* menekan tombol pada saat kondisi *text box* kosong maka akan ada validasi *title is required* pada *title* dan *password atleast have 8 character long* pada *password* dan *confirm password*. Serta terdapat validasi ketika *password* dan *confirm password* tidak sama yaitu *password do not match*.



Gambar 3.24. Tampilan *Create Document*



Gambar 3.25. Tampilan *Create Document - Password Kosong*



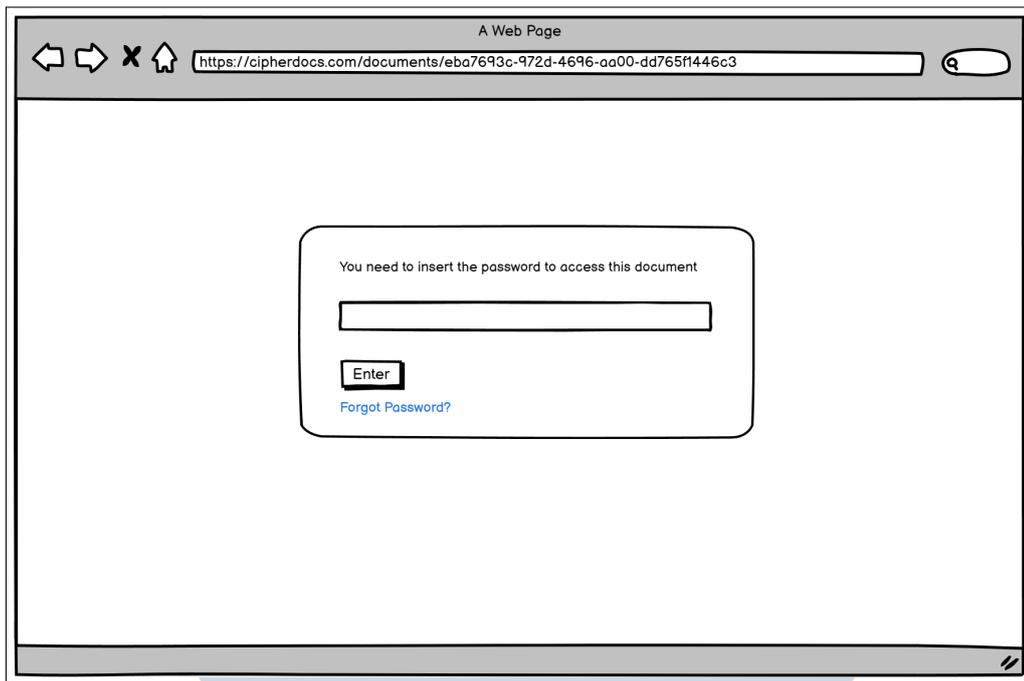
Gambar 3.26. Tampilan *Create Document - Password Tidak Sesuai*

E Tampilan *Insert Password*

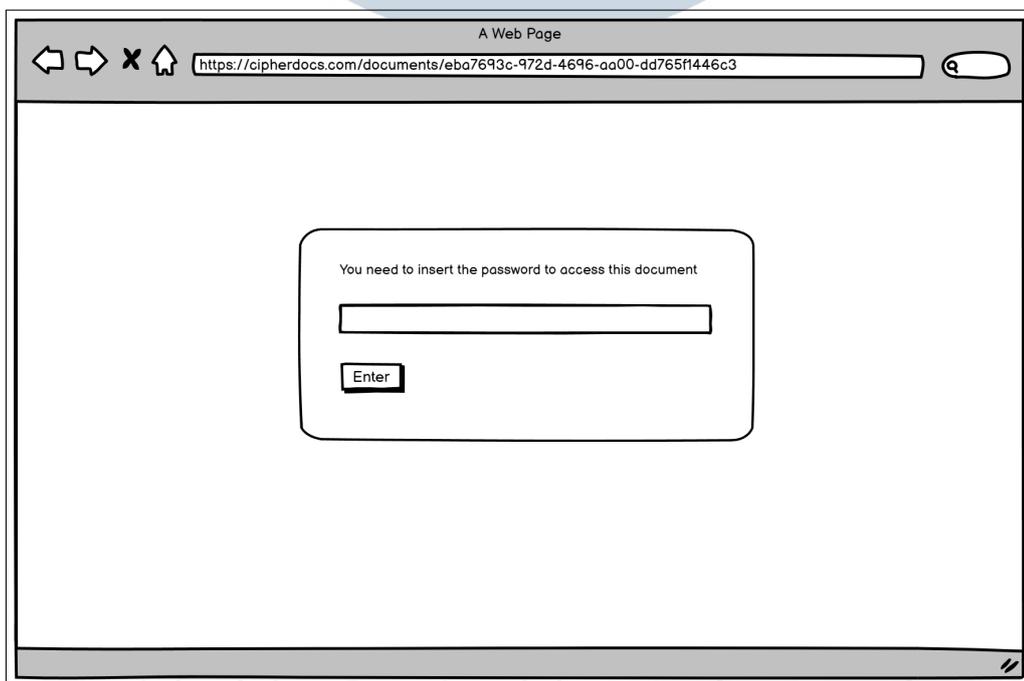
Gambar 3.27 merupakan rancangan antarmuka ketika *user* mengakses dokumen miliknya. Gambar 3.28 ketika *user* mengakses dokumen milik *user lain*. Gambar 3.29 merupakan rancangan antarmuka validasi pada saat input yang diberikan kosong pada saat mengakses dokumen milik *user sendiri*. Gambar 3.30 merupakan rancangan tampilan validasi pada saat *text box* kosong saat *user* ingin mengakses dokumen *user lain*.

Pada tampilan ini terdapat satu buah *header* yang memberikan informasi kepada *user* bahwa *user* harus memasukkan *password* untuk dapat mengakses dokumen yang sudah dipilih. Serta terdapat satu buah *text box* untuk memasukkan *password* dari dokumen. Terdapat satu buah tombol *enter* yang berfungsi untuk sistem melakukan validasi terhadap *password* yang diinput.

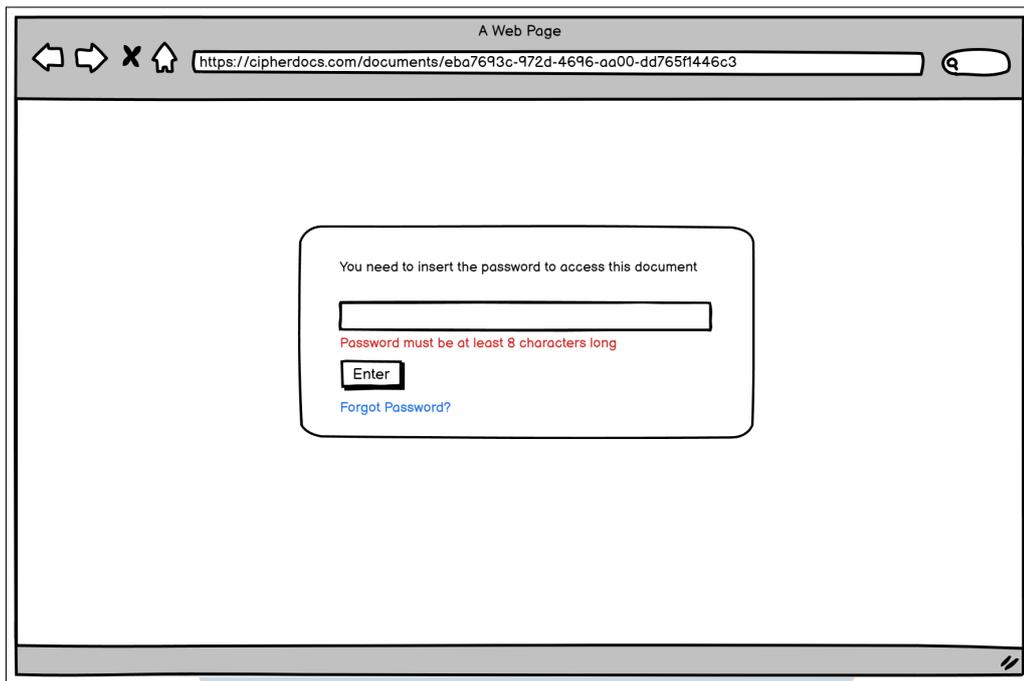
Pada halaman ini juga terdapat satu buah *hyperlink* yaitu *forgot password* yang dimana hanya akan tampil ketika *user* yang sedang *sign in* merupakan pemilik dari dokumen yang diakses. Pada saat *user* mengakses dokumen orang lain *hyperlink* tersebut tidak akan tampil. Terdapat dua validasi pada *text box password* yaitu ketika *password* kosong dan *password* tidak sesuai yang akan tampil dibawah *text box* dengan warna merah.



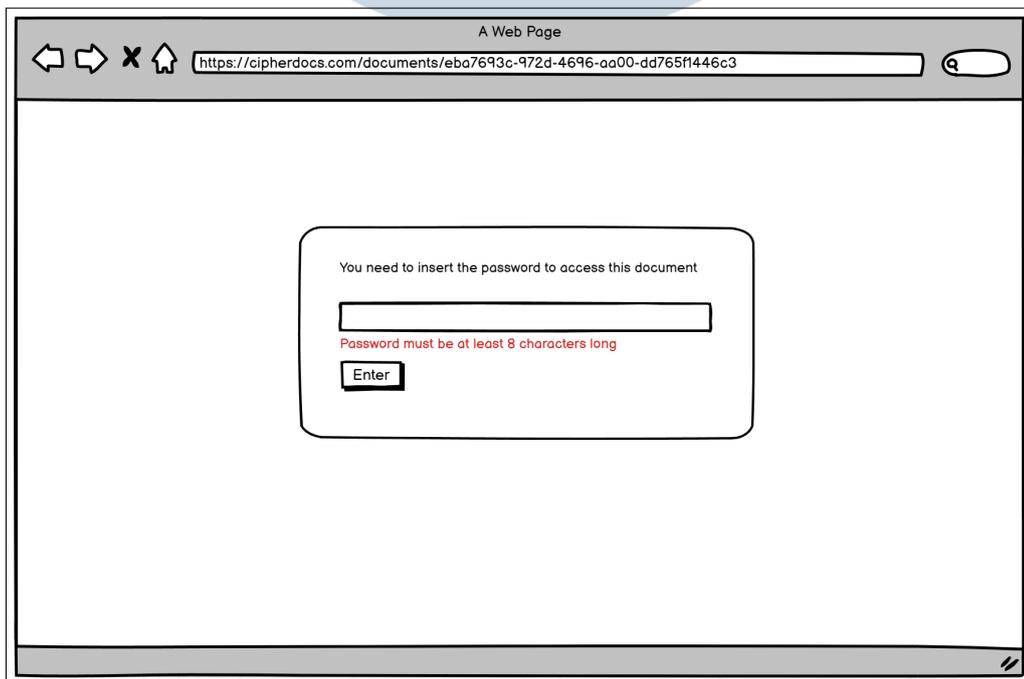
Gambar 3.27. Tampilan *Insert Password* - Pemilik



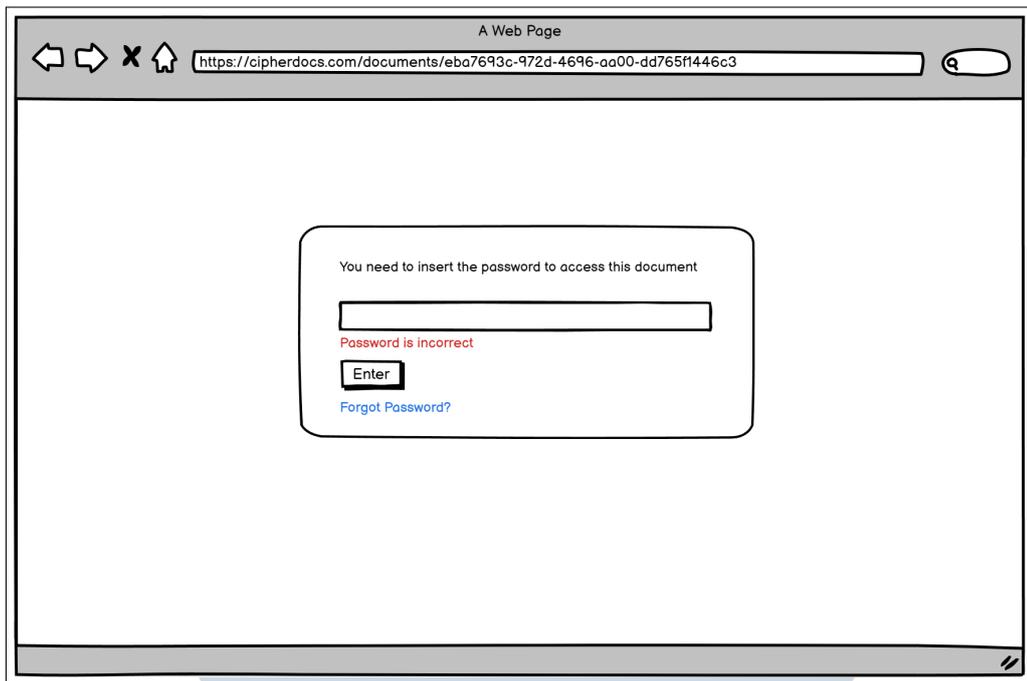
Gambar 3.28. Tampilan *Insert Password* - Bukan Pemilik



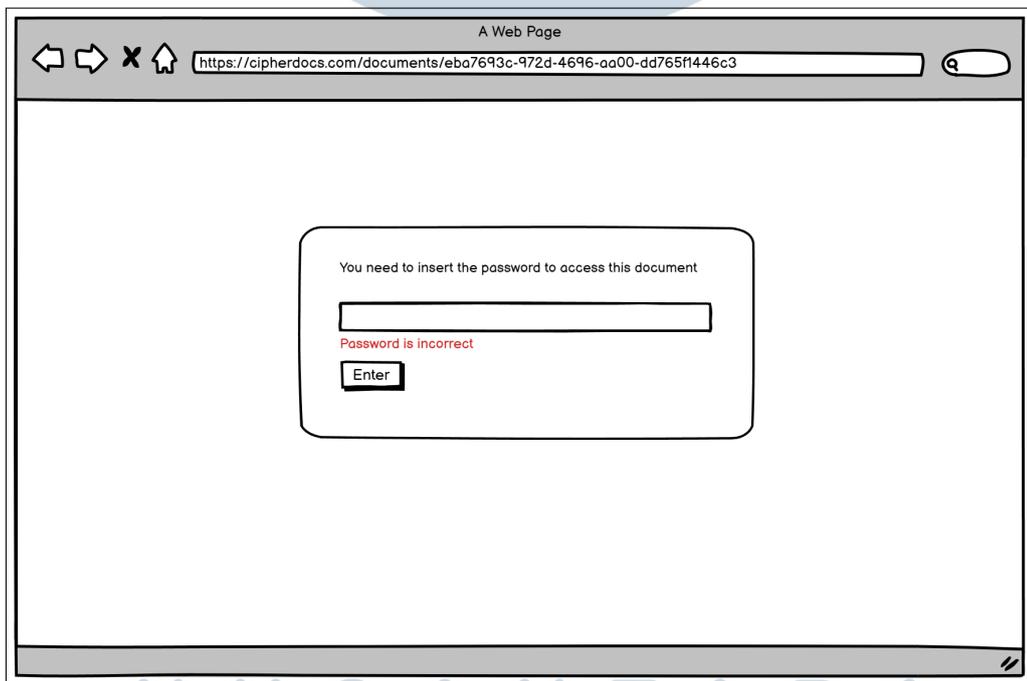
Gambar 3.29. Tampilan *Insert Password Kosong* - Pemilik



Gambar 3.30. Tampilan *Insert Password Kosong* - Bukan Pemilik



Gambar 3.31. Tampilan *Insert Password* Salah - Pemilik

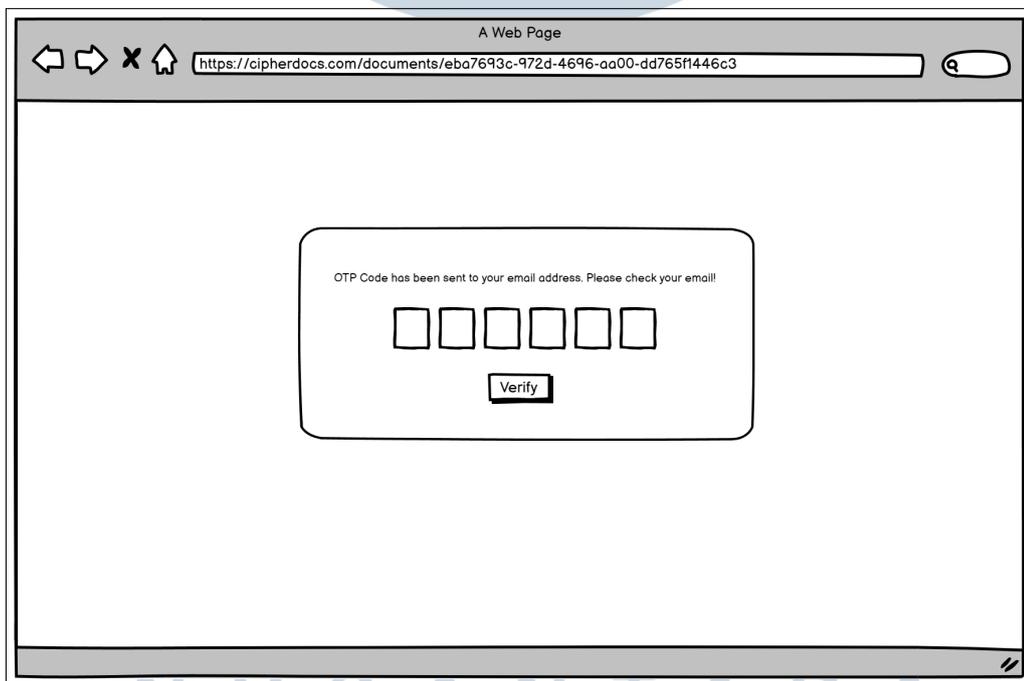


Gambar 3.32. Tampilan *Insert Password* Salah - Bukan Pemilik

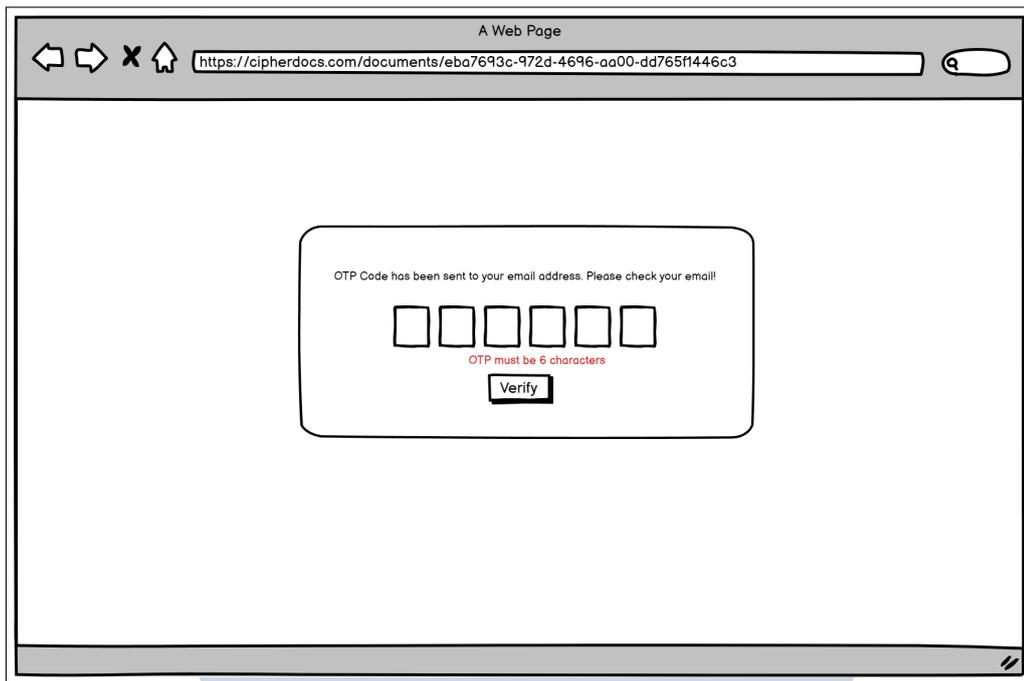
F Tampilan *Send OTP*

Gambar 3.33 merupakan tampilan rancangan antarmuka verifikasi *one time password* (OTP). Gambar 3.34 merupakan rancangan tampilan dari validasi pada saat OTP kosong. Gambar 3.35 merupakan rancangan tampilan validasi ketika OTP yang dimasukkan tidak sesuai. Terdapat satu buah judul yang memberi informasi pada *user* bahwa OTP sudah dikirimkan melalui *email* dan meminta *user* untuk memeriksa *email* miliknya.

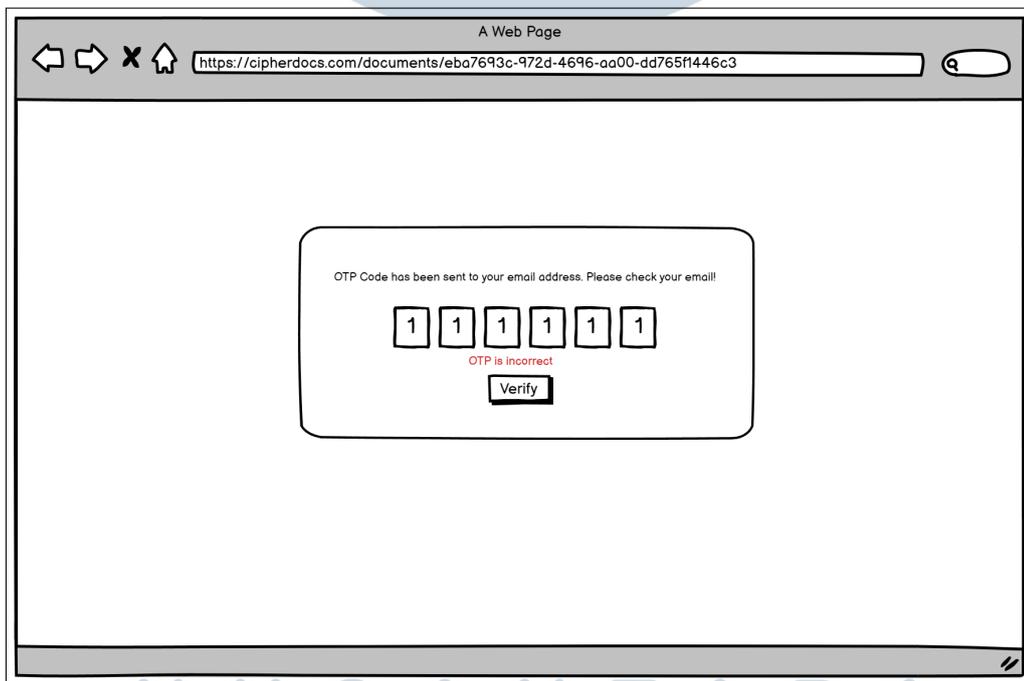
Terdapat enam buah *text box* yang berfungsi untuk menerima input kode OTP dari *user*. Serta terdapat tombol *verify* yang berfungsi untuk sistem melakukan verifikasi OTP. Terdapat satu buah teks tambahan untuk validasi dibawah *text box* berwarna merah. Pada saat *user* menekan tombol *verify* saat input yang diberikan masih dalam kondisi kosong atau tidak lengkap sehingga akan tampil validasi OTP *must be 6 characters*. Validasi selanjutnya merupakan pada saat OTP sudah lengkap tetapi tidak sesuai maka akan tampil pesan eror berupa *OTP is incorrect*. Tampilan *send OTP* akan tampil pada saat *user* memasukkan *password* dokumen dengan benar dan pada saat *user* mengakses halaman *forgot password*.



Gambar 3.33. Tampilan *Send OTP* - Verifikasi OTP



Gambar 3.34. Tampilan *Send OTP* - Verifikasi OTP Kosong

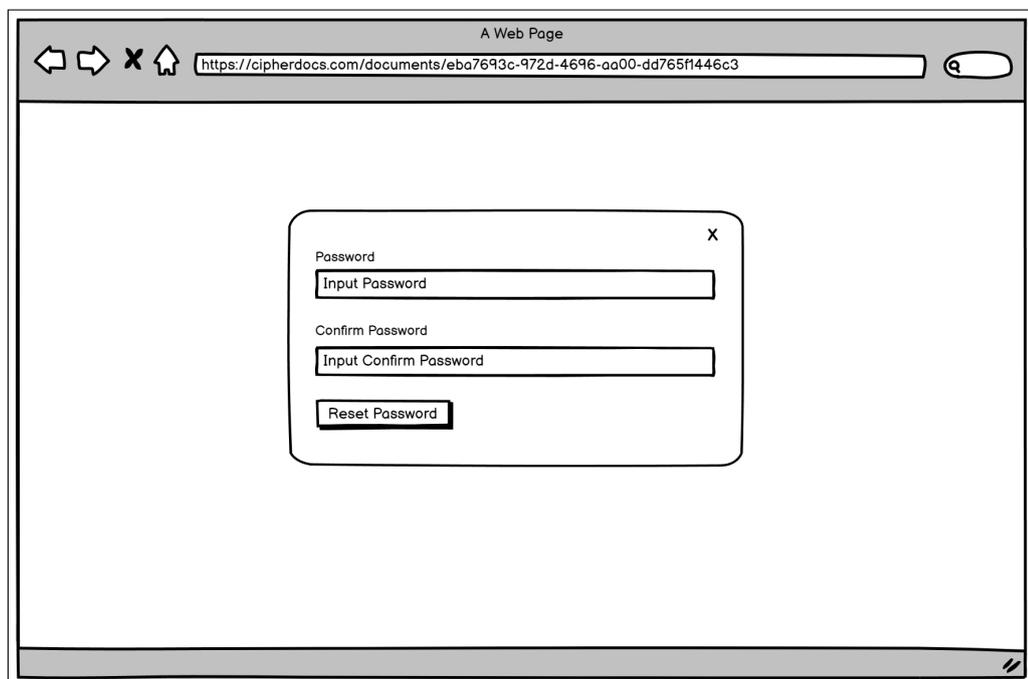


Gambar 3.35. Tampilan *Send OTP* - Verifikasi OTP Salah

G Tampilan *Forgot Password*

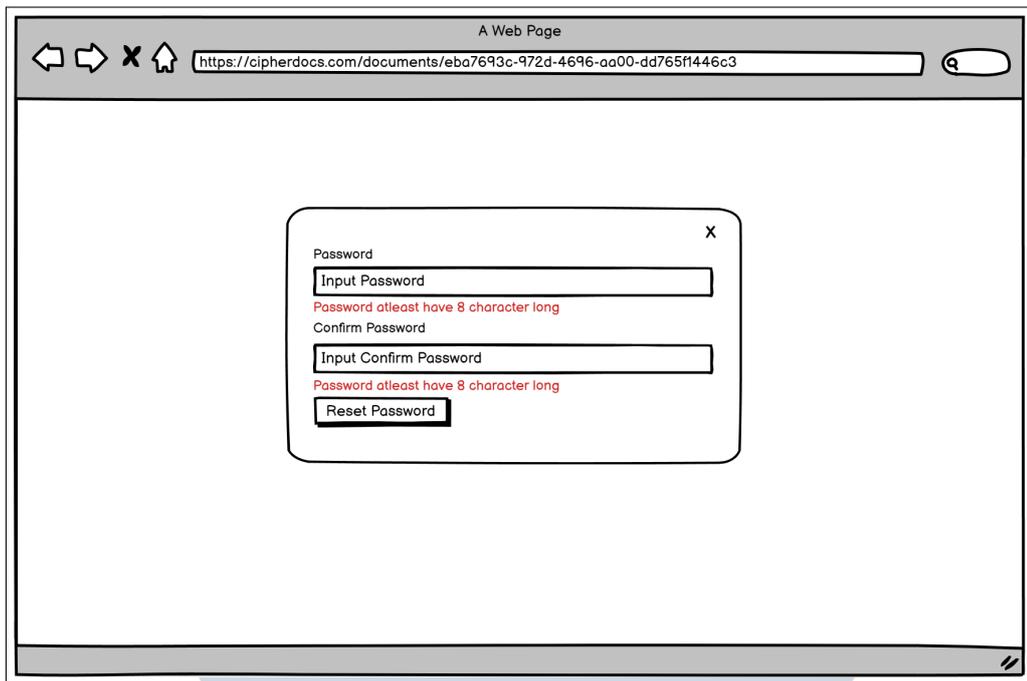
Gambar 3.36 merupakan rancangan tampilan dari *forgot password*. Gambar 3.37 merupakan rancangan tampilan antarmuka pada kondisi input yang dibutuhkan kosong. Gambar 3.38 merupakan rancangan tampilan antarmuka ketika *confirm password* tidak sama dengan *password*. Tampilan ini merupakan tampilan yang akan tampil setelah *one time password* atau OTP yang dimasukkan oleh *user* sesuai.

Terdapat dua buah *text box* dan dua buah *label* yang menunjukkan fungsi *text box*. *Text box* pada tampilan ini berfungsi untuk *user* memasukkan *password* baru dan *confirm password* untuk mencegah kesalahan pada *password* baru. Terdapat satu buah tombol *reset password* yang akan membuat sistem merubah atau mengganti *password* dari dokumen saat ini dan mengarahkan *user* ke halaman *edit document*. Serta terdapat dua buah teks tambahan yang menunjukkan validasi atau kesalahan yang dilakukan oleh *user*. Dimana pada saat *user* menekan tombol pada saat kondisi *password* dan *confirm password* kosong dan akan menampilkan validasi *password atleast have 8 character long*. Validasi kedua adalah pada saat *confirm password* yang diinput oleh *user* tidak sama dengan input *password* milik *user*.

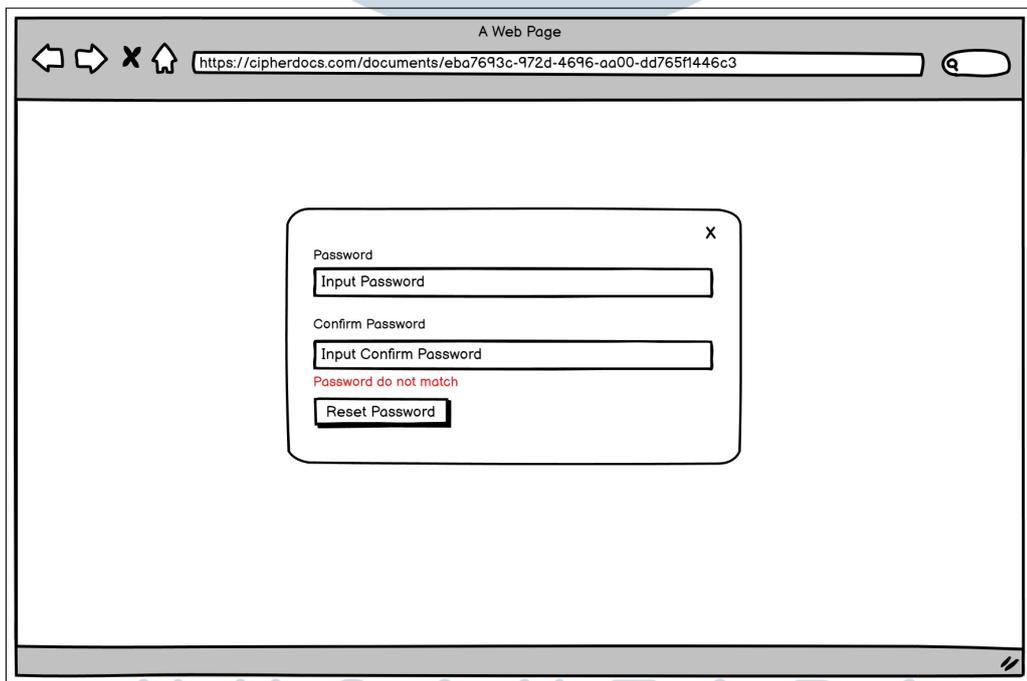


The image shows a web browser window with a title bar that says "A Web Page". The address bar contains the URL "https://cipherdocs.com/documents/eba7693c-972d-4696-aa00-dd765f1446c3". The main content area displays a form titled "Password" with a close button (X) in the top right corner. The form contains two input fields: "Input Password" and "Input Confirm Password". Below these fields is a button labeled "Reset Password".

Gambar 3.36. Tampilan *Forgot Password* - *Reset Password*



Gambar 3.37. Tampilan *Forgot Password - Reset Password* Kosong

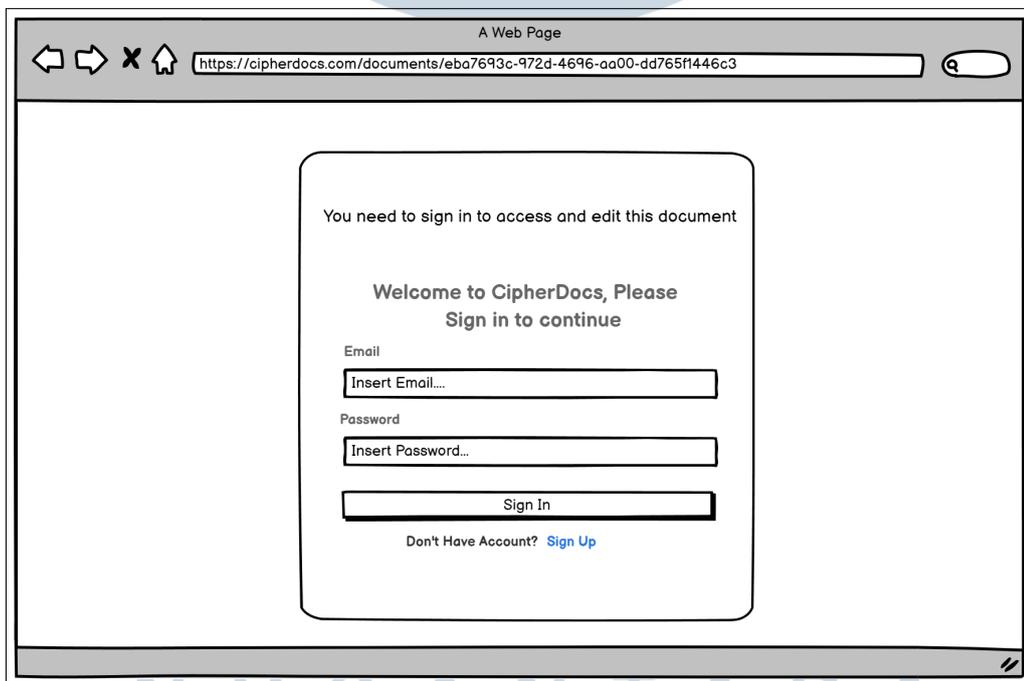


Gambar 3.38. Tampilan *Forgot Password - Reset Password* Tidak Sesuai

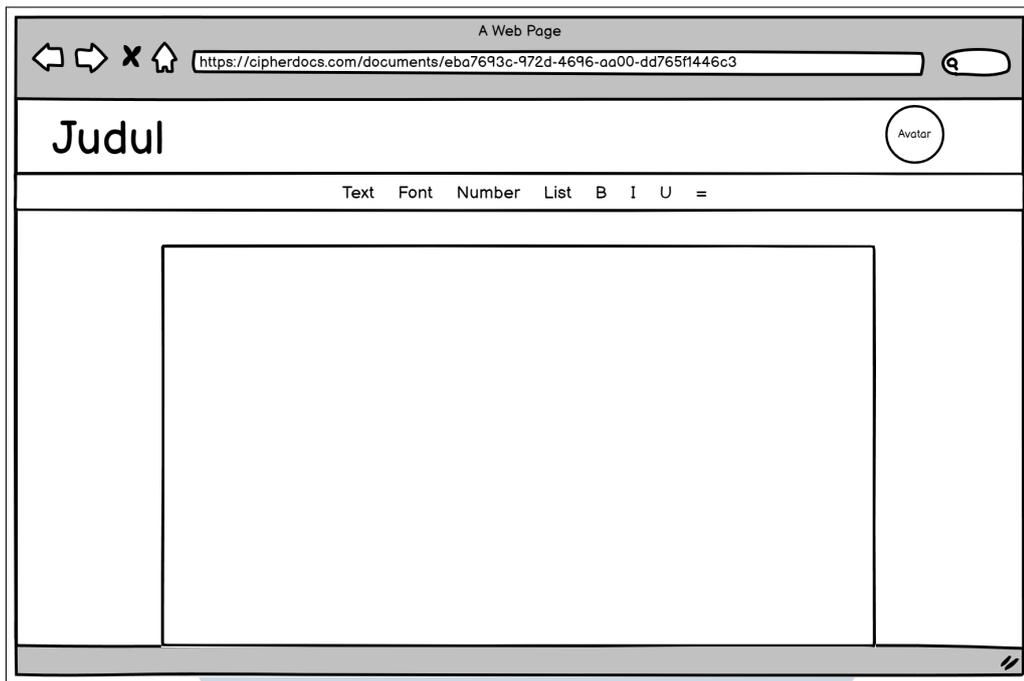
H Tampilan Document

Gambar 3.39 merupakan rancangan tampilan antarmuka pada saat *user* tanpa melakukan *sign in* terlebih dahulu. Gambar 3.40 merupakan rancangan tampilan antarmuka pada dokumen. Gambar 3.41 merupakan rancangan tampilan antarmuka pada saat *user* melakukan perubahan pada judul dokumen. Terdapat tampilan *pop up* pada saat ketika *user* tidak melakukan *sign in*. Pada halaman ini terdapat informasi bahwa *user* harus melakukan *sign in* terlebih dahulu untuk dapat mengakses dokumen. Serta akan tampil tampilan *sign in* yang sudah dijelaskan sebelumnya untuk meminta *user* untuk melakukan *sign in*.

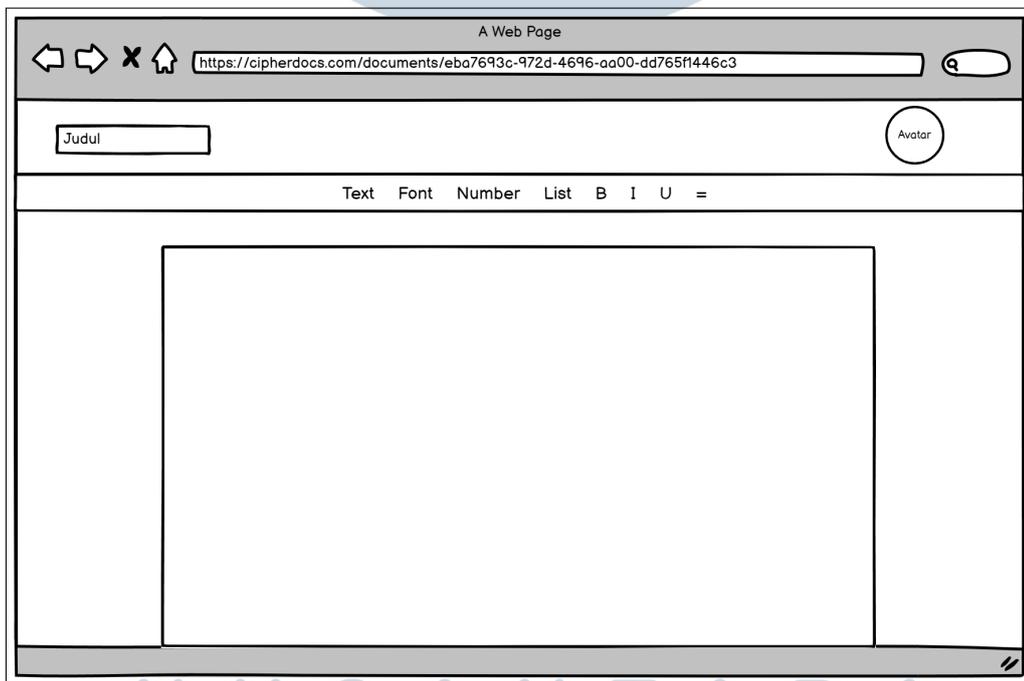
Pada tampilan dokumen terdapat judul dokumen pada pojok kiri atas halaman. Serta pada pojok kanan atas terdapat satu buah *image* atau avatar. Tampilan judul dapat berubah pada saat *user* menekannya dan sebelumnya berupa *text* menjadi sebuah *text box*. Serta terdapat beberapa fungsi pada *toolbox* untuk menentukan *heading*, *font*, *list number*, *list*, *bold*, *italic*, *underline*, dan lainnya. Terakhir yaitu terdapat dokumen atau halaman untuk *user* dapat mengolah kata atau membuat dokumen miliknya.



Gambar 3.39. Tampilan Document - Belum Sign In

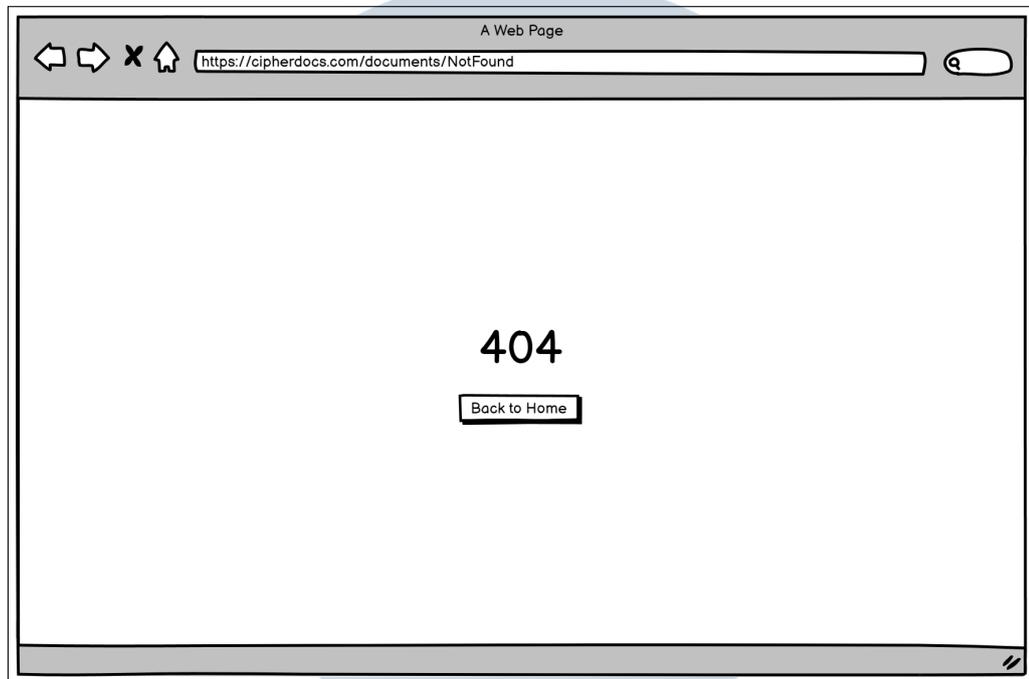


Gambar 3.40. Tampilan *Document*



Gambar 3.41. Tampilan *Document - Edit Title*

I Tampilan *Not Found*



Gambar 3.42. Tampilan *Not Found*

Gambar 3.42 merupakan rancangan tampilan antarmuka dari tampilan halaman yang tidak tersedia. Halaman ini akan ditampilkan pada saat *user* memasukkan *url* dokumen secara acak sehingga dokumen tidak ditemukan. Terdapat satu buah *header* yaitu 404 yang melambangkan halaman tidak ditemukan. Serta terdapat satu buah *button back to home* yang berfungsi untuk sistem mengarahkan *user* kembali ke halaman *home*.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A