

BAB IV

PERANCANGAN

Pada bab ini, peneliti akan menjelaskan secara detail bagaimana kebutuhan spesifikasi sistem, cara kerja sistem, rancangan antarmuka sistem dan rancangan pengujian sistem yang akan dibuat.

4.1 Analisis Sistem Terdahulu

Analisis sistem terdahulu bertujuan untuk memahami, mengidentifikasi kelebihan dan kekurangan dari sistem terdahulu. Pada tugas akhir ini peneliti mencari situs perbandingan dan belum menemukan situs untuk pencarian jenis bunga mawar, kemudian peneliti memilih jurnal yang berjudul “Purwarupa Sistem *Content Based Image Retrieval* Untuk Pencarian Produk Sepatu” (Baldri et al., 2018) sebagai perbandingan.

4.2 Spesifikasi Kebutuhan Sistem Baru

Spesifikasi kebutuhan sistem baru ini membagi masalah spesifikasi kebutuhan sistem menjadi empat bagian, yaitu spesifikasi kebutuhan perangkat lunak, perangkat keras, input, dan output.

4.2.1 Spesifikasi Kebutuhan Perangkat Lunak

Spesifikasi kebutuhan perangkat lunak merupakan hal yang penting dalam mempersiapkan penelitian. Peneliti akan menjabarkan kebutuhan, mulai dari perangkat sampai aplikasi editor. Berikut merupakan spesifikasi kebutuhan perangkat lunak.

Tabel 4.1 Spesifikasi Kebutuhan Perangkat Lunak

No	Perangkat	Keterangan
1	Windows 10 64-bit	Sistem operasi komputer yang digunakan
2	Visual Studio Code	<i>Platform</i> editor yang digunakan untuk membuat kode program aplikasi
3	PhpMyAdmin	<i>Platform</i> yang digunakan untuk mengelola <i>database</i>
4	XAMPP	<i>Platform</i> yang digunakan untuk menjalankan <i>local web server</i>
5	<i>Web Browser</i>	<i>Platform</i> yang digunakan untuk mengakses <i>web</i> di internet.
6	Figma	<i>Platform</i> yang digunakan untuk desain tampilan aplikasi

4.2.2 Spesifikasi Kebutuhan Perangkat Keras

Spesifikasi kebutuhan perangkat keras yang digunakan pada penelitian ini mencakup prosesor, harddisk dan memory. Terlihat spesifikasi kebutuhan perangkat keras tertera pada table berikut ini.

Tabel 4.2 Spesifikasi Kebutuhan Perangkat Keras

No	Perangkat Keras	Spesifikasi
1	<i>Processor</i>	Intel i3-7020u
2	<i>Harddisk</i>	256 GB SSD & 1 TB HDD
3	<i>RAM</i>	8 GB RAM

4.2.3 Spesifikasi Kebutuhan Input

Aplikasi pencarian jenis bunga mawar pada penelitian ini, tentu diperlukan proses input agar aplikasi dapat berjalan dengan baik. Proses input yang dibutuhkan pada aplikasi ini sebagai berikut.

1. Input Gambar

Untuk menggunakan aplikasi ini, pengguna harus melakukan input gambar yang ingin dicari untuk melakukan pencarian gambar.

2. Input Data Gambar

Untuk menggunakan aplikasi ini, pengguna harus melakukan input data gambar yang digunakan sebagai *dataset*. *Form* data gambar yang diinput pada halaman data gambar berupa nama gambar, gambar, dan keterangan. Gambar yang sudah di input akan ditampilkan pada list di halaman data gambar dan di tampilan hasil pencarian.

3. Input Data User

Admin dapat menambahkan *user* agar dapat mengelola *website*. *Form* data *user* yang diinput pada halaman data *user* berupa nama lengkap, dan email. *User* yang sudah di input akan ditampilkan pada list di halaman data *user*.

4.2.4 Spesifikasi Kebutuhan Output

Hasil dari aplikasi pencarian jenis bunga mawar fokus kepada hasil gambar. Berikut merupakan spesifikasi output aplikasi.

1. Gambar yang ditampilkan akan berukuran 1000×1000 piksel.
2. Gambar dengan jarak *euclidean* yang mirip dengan gambar yang di *upload*.

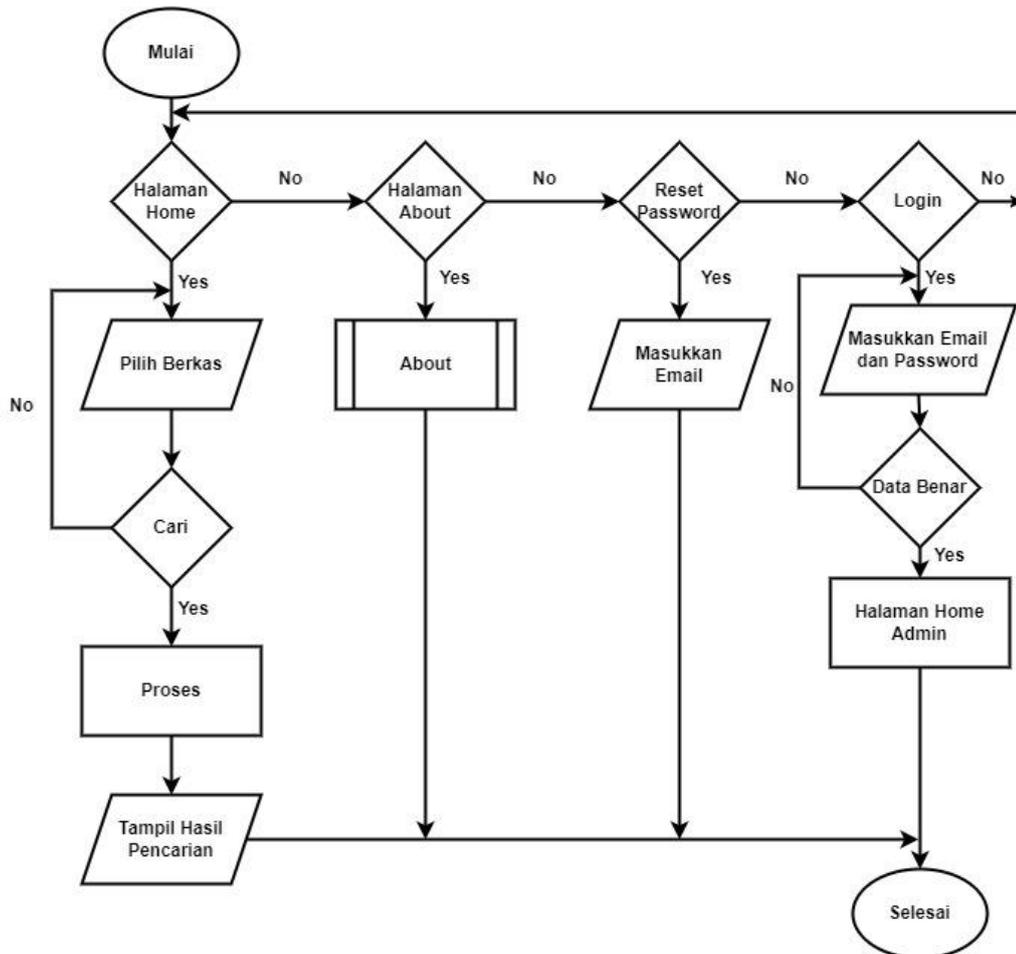
3. Gambar yang telah ditambahkan akan ditampilkan pada halaman data gambar.
4. User yang telah ditambahkan akan ditampilkan pada halaman data user.

4.3 Perancangan Sistem

Perancangan sistem bertujuan untuk menganalisis kebutuhan sistem agar dapat membangun aplikasi secara menyeluruh. Proses perancangan sistem ini melibatkan penjabaran yang detail mengenai prosedur langkah demi langkah, desain, dan perancangan pengujian aplikasi. Berbagai jenis diagram seperti *flowchart*, *use case diagram*, *skenario use case*, *activity diagram*, *sequence diagram* dan desain antarmuka digunakan dalam perancangan sistem untuk mempermudah pemahaman dan visualisasi proses yang terlibat.

4.3.1 Flowchart

Flowchart adalah suatu diagram yang berguna dalam merepresentasikan urutan langkah-langkah dan keputusan yang dibutuhkan dalam menjalankan suatu proses atau aktivitas dalam sistem tertentu. *Flowchart* memiliki peran penting dalam memvisualisasikan secara jelas dan sistematis alur kerja atau proses yang terjadi dalam suatu sistem atau aktivitas.

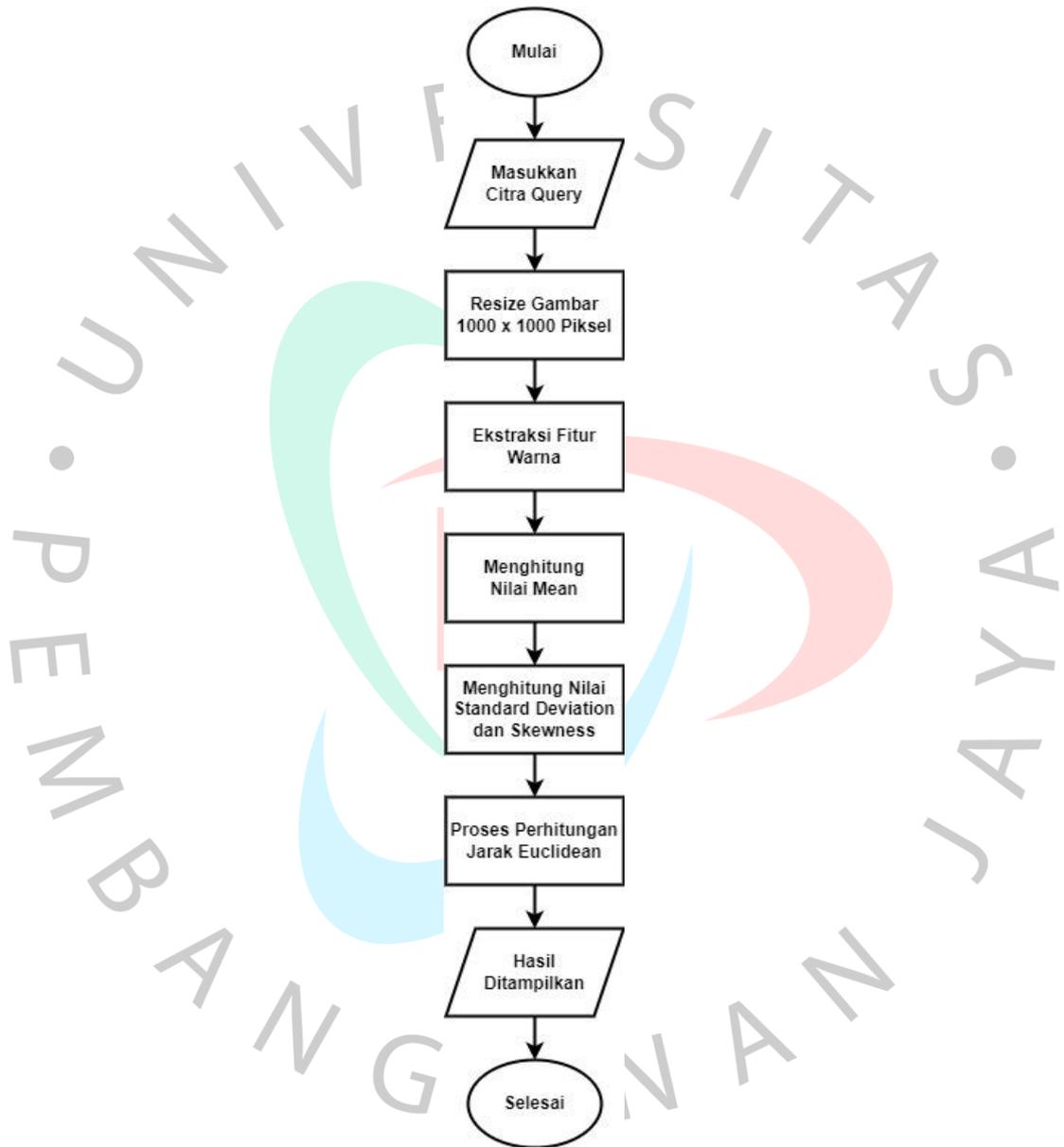


Gambar 4.1 Flowchart

Gambar 4.3 diatas merupakan *flowchart* pada aplikasi *website* pencarian jenis bunga mawar. Penjelasan pada *flowchart* diatas akan dijelaskan sebagai berikut.

1. User dapat mencari gambar pada halaman home. Langkah pertama, user harus *input* gambar bunga mawar yang ingin dicari. Kemudian, sistem akan memproses gambar tersebut dan sistem akan menampilkan gambar yang mirip dengan gambar yang dicari pada tampilan hasil pencarian.
2. User dapat masuk ke halaman about. User dapat memilih menu about. Menu tersebut akan menampilkan informasi mengenai *website* pencarian jenis bunga mawar.
3. User dapat melakukan *reset password* jika user telah memiliki *account*. Langkah pertama, user harus memiliki *account* terlebih dahulu pada *website* tersebut. Kemudian, *input* email yang terdaftar dan sistem akan mengirim *password* baru melalui email.

4. User dapat melakukan *login* jika telah memiliki *account* untuk masuk ke *website* admin. Langkah pertama, admin harus *input* email dan *password* dengan benar agar dapat masuk ke dalam *website* admin. User akan langsung disajikan dengan tampilan *website* admin dengan menu tambahan.



Gambar 4.2 *Flowchart Content Based Image Retrieval*

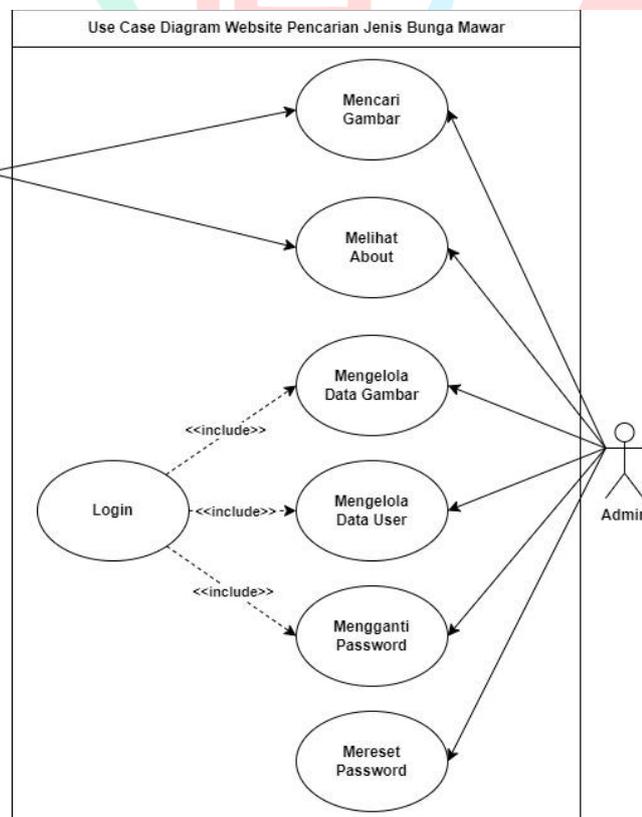
Gambar diatas merupakan alur proses dari *Content Based Image Retrieval (CBIR)*. Penjelasan pada *flowchart* diatas akan dijelaskan sebagai berikut.

1. Mulai,
2. Masukkan citra sebagai *query*,

3. Setelah citra dimasukkan, gambar tersebut akan di resize menjadi 1000×1000 piksel,
4. Selanjutnya akan dilakukan ekstraksi fitur warna untuk mendapatkan nilai warna pada gambar,
5. Selanjutnya hitung nilai *mean*,
6. Kemudian hitung nilai *standard deviation* dan *skewness*,
7. Setelah nilai *mean*, *standard deviation*, dan *skewness* diketahui, maka proses selanjutnya akan menghitung jarak *euclidean* untuk mengetahui seberapa jarak antara nilai citra *query* dengan citra *database*,
8. Setelah proses menghitung jarak *euclidean* citra *query* dengan citra *database* selesai maka hasil citra gambar akan ditampilkan,
9. Selesai.

4.3.2 Use Case Diagram

- *Use case diagram* dibawah ini merupakan alur dari proses aktivitas antara pengguna dengan sistem aplikasi *web* pencarian gambar. Berikut adalah penjelasannya.



Gambar 4.3 Use Case Diagram

Gambar *use case diagram* diatas memiliki dua pengguna yang dapat mengakses aplikasi pencarian jenis bunga mawar yaitu *user* dan *admin*.

- a) *User* dapat mencari gambar dan melihat informasi pada halaman *about*.
- b) *Admin* dapat melakukan aktivitas yang dilakukan *user* dan *admin* dapat mengelola data. Data tersebut yaitu mengelola data gambar seperti menambah, mengedit, dan menghapus gambar. *Admin* juga dapat mengelola data *user* untuk menambah dan menghapus *user*. Selain itu, *admin* juga dapat mengganti *password*. Untuk melakukan tiga aktivitas tersebut *admin* diharuskan untuk melakukan *login* terlebih dahulu. Namun, untuk mereset *password* dapat dilakukan tanpa harus *login*.

4.3.2.1 Skenario Use Case Diagram

Skenario *use case* adalah untuk menggambarkan dengan jelas bagaimana pengguna atau aktor akan menggunakan sistem dalam situasi nyata guna memastikan bahwa sistem tersebut dapat berfungsi dengan efektif dan efisien. Berikut merupakan penjelasan dari *use case diagram* yang dijelaskan menggunakan tabel skenario.

Tabel 4.3 Skenario Use Case Login

ID	UC7
Nama use case	<i>Login</i> .
Aktor	<i>Admin</i> .
Deskripsi	<i>Admin</i> masuk dengan cara <i>login</i> untuk mengakses data.
<i>Pre-condition</i>	-
Langkah-langkah	<i>Admin</i> memasukkan <i>email</i> dan <i>password</i> .
<i>Trigger</i>	Ketika ingin mengelola data gambar, mengelola data <i>user</i> , dan mengganti <i>password</i> .
<i>Post-Condition</i>	Berhasil <i>login</i> dan masuk ke tampilan <i>website</i> dengan menu tambahan.

Tabel 4.4 Skenario Use Case Mencari Gambar

ID	UC1
Nama use case	Mencari Gambar.
Aktor	<i>User</i> .
Deskripsi	<i>User</i> memasukkan file gambar untuk mencari gambar.
<i>Pre-condition</i>	-

Langkah-langkah	User memasukkan file gambar yang ingin dicari.
Trigger	Ketika ingin mencari gambar.
Post-Condition	Berhasil mencari gambar dan menampilkan tampilan hasil pencarian gambar.

Tabel 4.5 Skenario Use Case Melihat About

ID	UC2
Nama use case	Melihat About.
Aktor	User.
Deskripsi	User dapat ke halaman about untuk melihat informasi website.
Pre-condition	-
Langkah-langkah	User klik menu about.
Trigger	Ketika ingin ke halaman about untuk melihat informasi website.
Post-Condition	Berhasil masuk ke halaman about.

Tabel 4.6 Skenario Use Case Mengelola Data Gambar

ID	UC3
Nama use case	Mengelola Data Gambar.
Aktor	Admin.
Deskripsi	Admin dapat mengelola data gambar.
Pre-condition	Login dan klik menu data gambar.
Langkah-langkah	<ol style="list-style-type: none"> 1. Menambah data gambar dengan klik button tambah. 2. Mengedit data gambar dengan klik ikon pensil pada kolom aksi. 3. Menghapus data gambar dengan klik ikon tempat sampah pada kolom aksi.
Trigger	Ketika ingin mengelola data gambar.
Post-Condition	Berhasil mengelola data gambar.

Tabel 4.7 Skenario Use Case Mengelola Data User

ID	UC4
Nama use case	Mengelola Data User.
Aktor	Admin.
Deskripsi	Admin dapat mengelola data user.
Pre-condition	Login dan klik menu data user.

Langkah-langkah	<ol style="list-style-type: none"> 1. Menambah data user dengan klik button tambah. 2. Menghapus data user dengan klik ikon tempat sampah pada kolom aksi.
<i>Trigger</i>	Ketika ingin mengelola data user.
<i>Post-Condition</i>	Berhasil mengelola data user.

Tabel 4.8 Skenario Use Case Mengganti Password

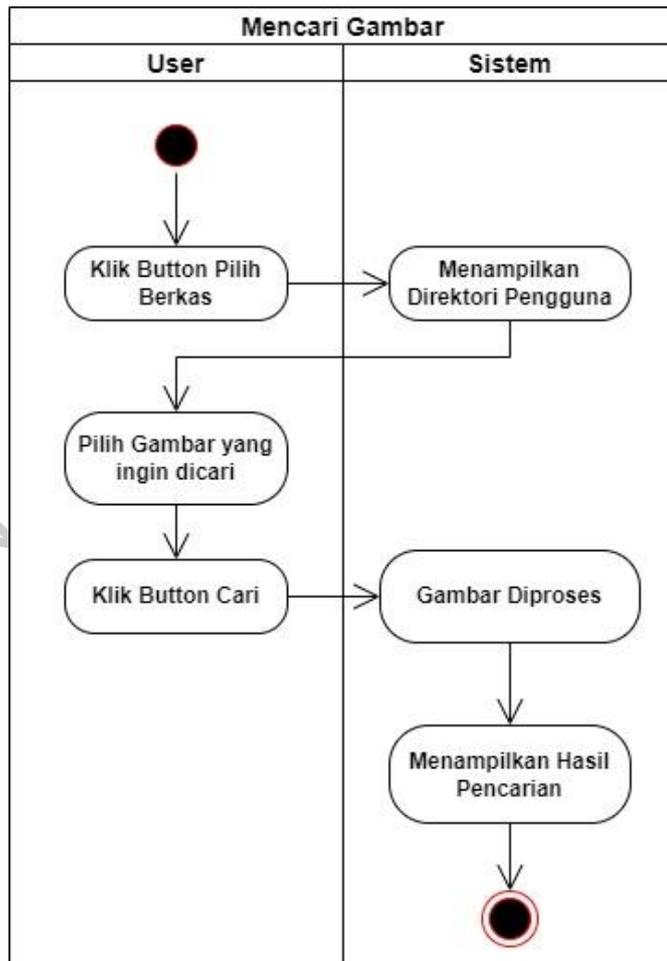
ID	UC5
Nama use case	Mengganti <i>Password</i> .
Aktor	Admin.
Deskripsi	Admin dapat mengganti <i>password</i> .
<i>Pre-condition</i>	<i>Login</i> dan klik ganti <i>password</i> .
Langkah-langkah	Admin memasukkan <i>password</i> baru.
<i>Trigger</i>	Ketika ingin mengganti <i>password</i> .
<i>Post-Condition</i>	Berhasil mengganti <i>password</i> .

Tabel 4.9 Skenario Use Case Mereset Password

ID	UC6
Nama use case	Mereset <i>Password</i> .
Aktor	Admin.
Deskripsi	Admin dapat melakukan <i>reset password</i> .
<i>Pre-condition</i>	-
Langkah-langkah	Admin memasukkan <i>email</i> .
<i>Trigger</i>	Ketika ingin melakukan <i>reset password</i> .
<i>Post-Condition</i>	Berhasil mereset <i>password</i> .

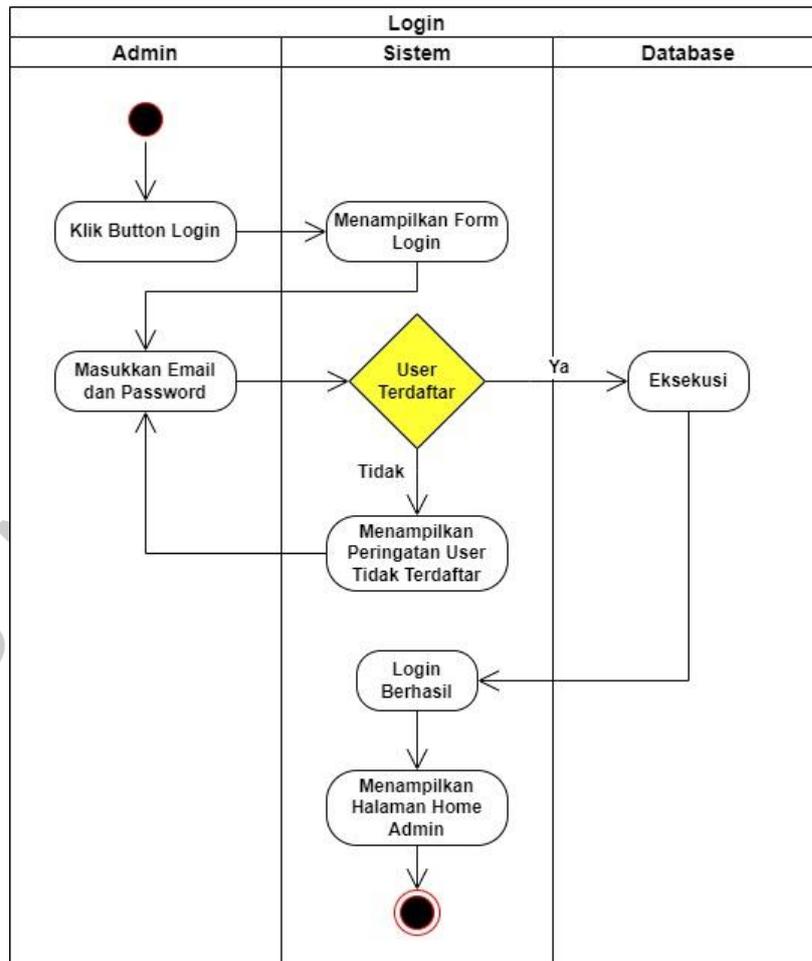
4.3.3 Activity Diagram

Selama proses pengembangan aplikasi, *activity diagram* digunakan untuk menjelaskan urutan dan sistematis dari aktivitas yang terlibat dalam proses yang telah dijelaskan pada *use case diagram* sebelumnya. Berikut adalah *activity diagram* pada perancangan aplikasi pencarian jenis bunga mawar.



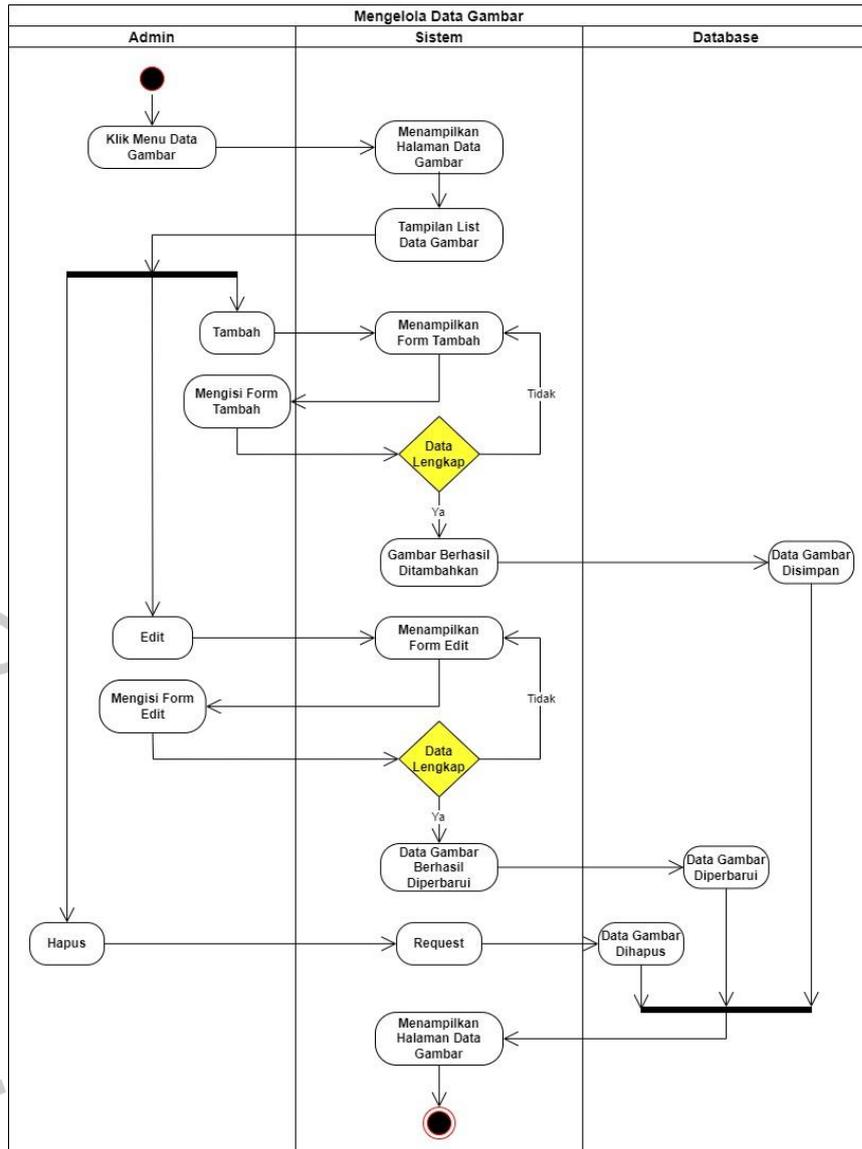
Gambar 4.4 Activity Diagram Mencari Gambar

Gambar diatas merupakan *activity diagram* mencari gambar. Pada halaman home *website* pencarian jenis bunga mawar, *user* klik *button* “pilih berkas” lalu sistem akan menampilkan direktori *user*. Selanjutnya *user* memilih gambar bunga mawar yang ingin dicari, kemudian klik *button* “cari”. Setelah itu, sistem akan memproses gambar tersebut dan sistem akan menampilkan hasil pencarian yang mirip dengan gambar yang dicari.



Gambar 4.5 Activity Diagram Login

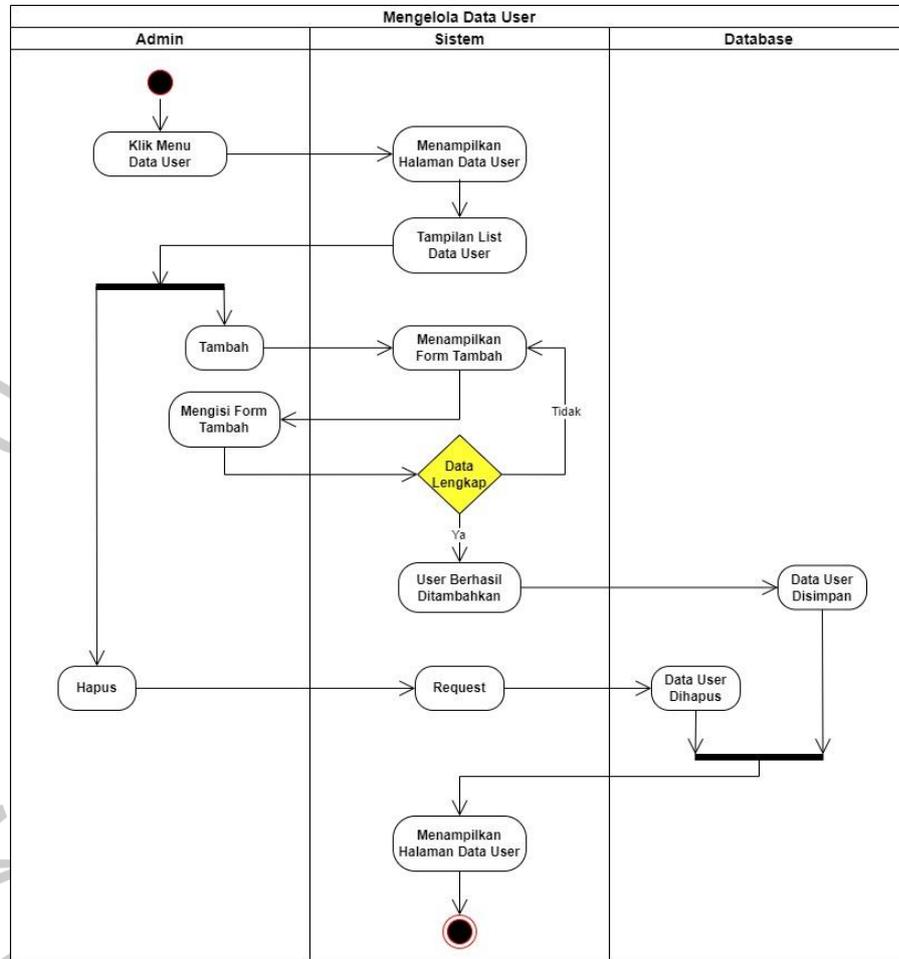
Gambar diatas merupakan *activity diagram login*. Pada halaman *home website* pencarian jenis bunga mawar, admin klik *button* “*login*”. Selanjutnya, form *login* akan ditampilkan lalu admin dapat memasukkan email dan *password* yang terdaftar. Sistem akan menampilkan peringatan atau pesan jika email dan *password* salah. Tetapi jika email dan *password* benar, maka sistem akan menampilkan pesan bahwa *login* berhasil dan akan masuk ke halaman *home admin*.



Gambar 4.6 Activity Diagram Mengelola Data Gambar

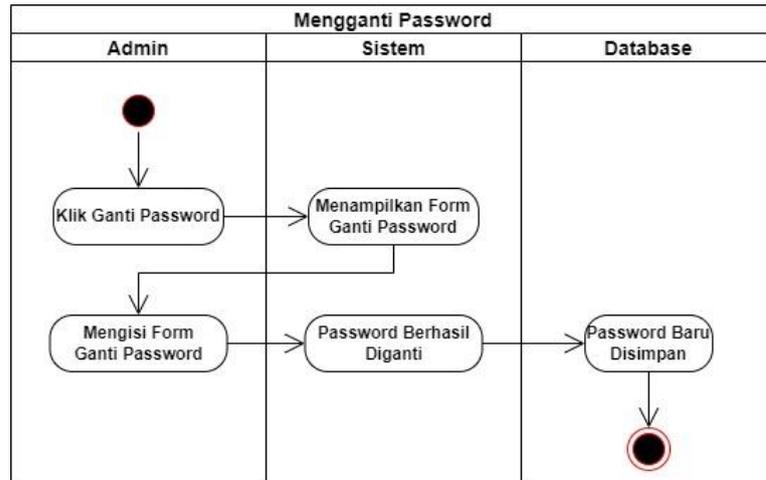
Gambar diatas merupakan *activity diagram* mengelola data gambar. Admin dapat mengelola data gambar setelah melakukan *login* untuk menampilkan menu tambahan pada *website*. Setelah itu, admin dapat klik menu “data gambar” dan sistem akan menampilkan halaman data gambar. Admin dapat menambah gambar dengan klik button “tambah”, kemudian mengisi form tambah. Jika data sudah benar maka data gambar akan disimpan di dalam *database* dan ditampilkan dilist data gambar. Selanjutnya admin dapat mengedit data gambar dengan klik “ikon pensil” pada kolom aksi, kemudian sistem akan menampilkan form edit gambar. Jika data sudah diperbarui dengan benar maka data gambar diperbarui di dalam *database* dan akan ditampilkan dilist data gambar. Kemudian admin juga dapat

menghapus data gambar dengan klik “ikon tempat sampah” pada kolom aksi, lalu data gambar yang dipilih maka akan terhapus dari tampilan dan dihapus dari *database*.



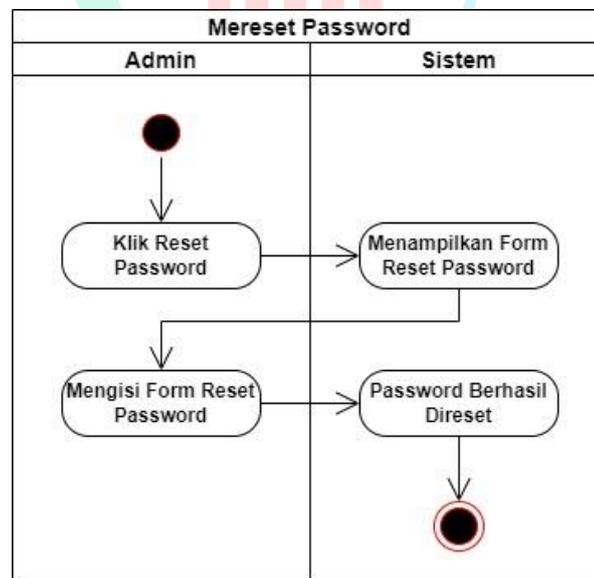
Gambar 4.7 Activity Diagram Mengelola Data User

Gambar diatas merupakan *activity diagram* mengelola data user. Admin dapat klik menu “data user” dan sistem akan menampilkan halaman data user. Admin dapat menambah user dengan klik *button* “tambah”, kemudian mengisi form tambah. Jika data sudah benar maka data user akan disimpan di dalam *database* dan ditampilkan dilist data user. Kemudian admin juga dapat menghapus data user dengan klik “ikon tempat sampah” pada kolom aksi, lalu data user yang dipilih maka akan terhapus dari tampilan dan dihapus dari *database*.



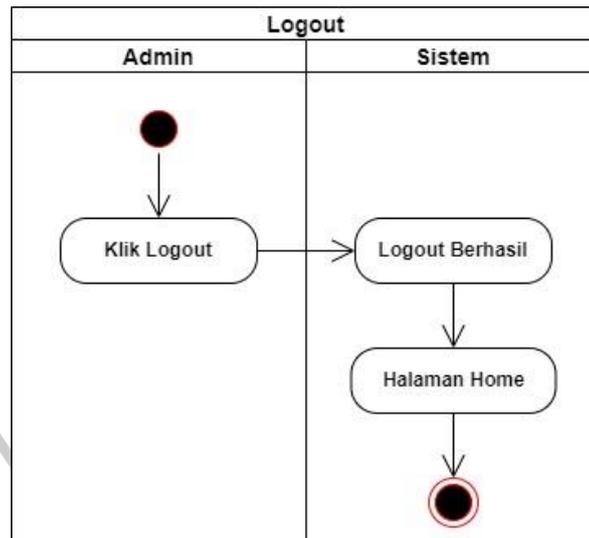
Gambar 4.8 Activity Diagram Mengganti Password

Gambar diatas merupakan *activity diagram* mengganti *password*. Admin dapat mengganti *password* dengan klik menu “ganti *password*”, kemudian sistem akan menampilkan form ganti *password*. Selanjutnya admin mengisi form tersebut dengan *password* baru. Jika sudah maka *password* yang ada di dalam *database* akan diperbarui.



Gambar 4.9 Activity Diagram Meraset Password

Gambar diatas merupakan *activity diagram* meraset *password*. Admin dapat *reset password* dengan klik menu “*reset password*”. Sistem akan menampilkan form *reset password* dan admin dapat mengisi form tersebut. Setelah itu, *password* baru akan dikirimkan oleh sistem ke email admin.

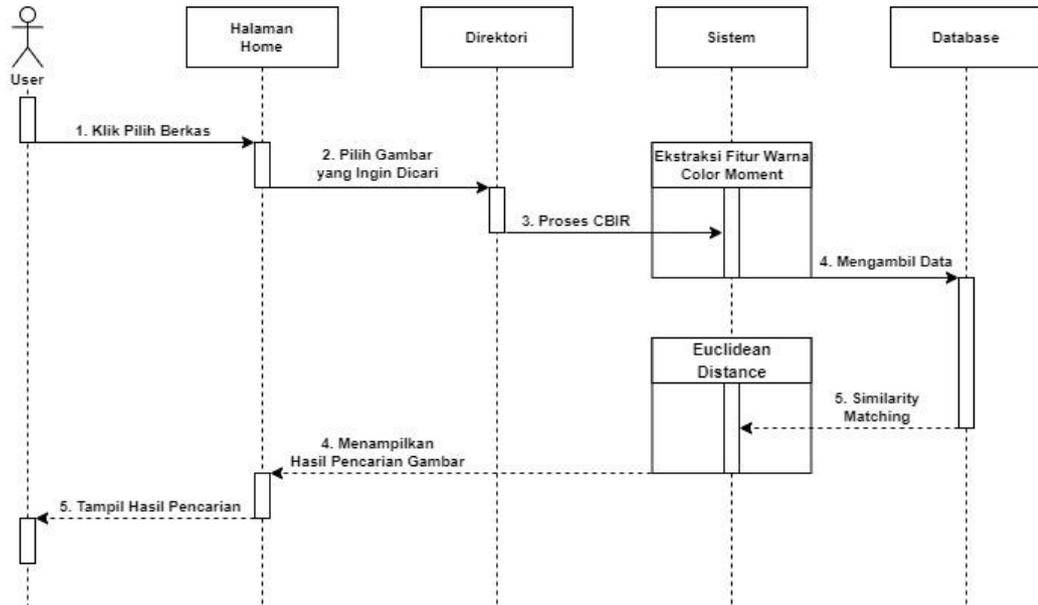


Gambar 4.10 Activity Diagram Logout

Gambar diatas merupakan *activity diagram logout*. Jika admin telah selesai melakukan aktivitas seperti mengelola data gambar dan data user, admin dapat keluar dengan klik *logout*. Kemudian, sistem akan kembali menampilkan tampilan halaman home untuk *user* dimana tidak ada menu data gambar, data user, dan ganti *password*.

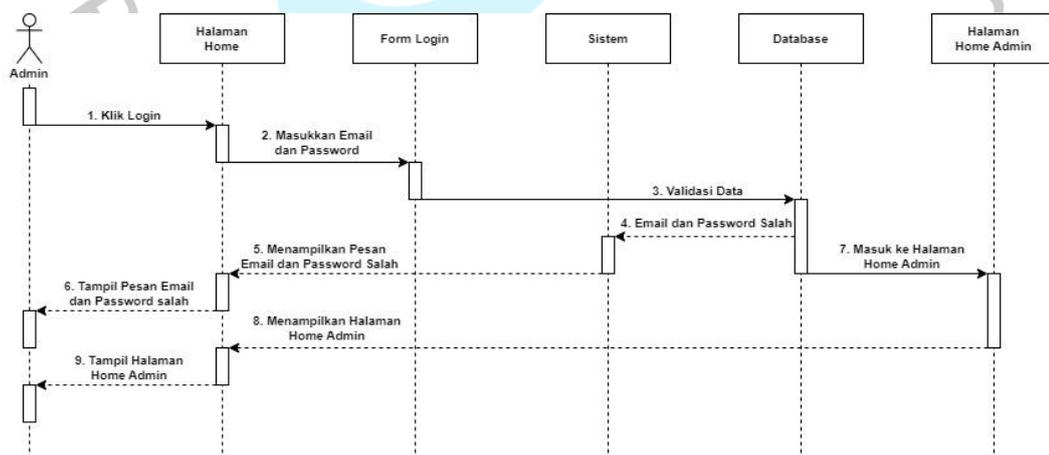
4.3.4 Sequence Diagram

Sequence diagram digunakan untuk memvisualisasikan urutan pesan atau interaksi yang dilakukan antara objek-objek yang terlibat dalam suatu proses atau kasus penggunaan. Hal ini bertujuan untuk memperlihatkan bagaimana interaksi antara objek-objek tersebut terjadi dan bagaimana keterkaitan antara objek-objek tersebut dalam suatu sistem. Berikut adalah *sequence diagram* pada perancangan aplikasi pencarian jenis bunga mawar.



Gambar 4.11 Sequence Diagram Mencari Gambar

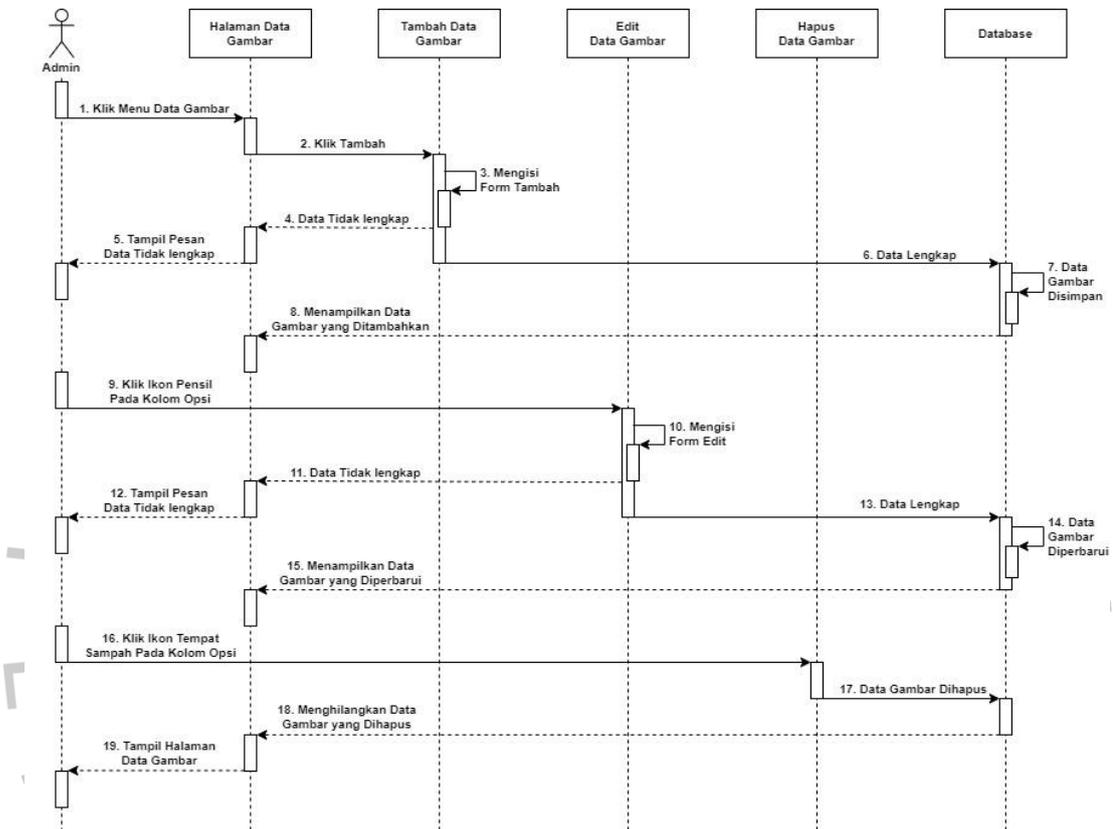
Sequence diagram diatas menjelaskan pengguna yang ingin mencari gambar. Pengguna dapat klik “pilih berkas” pada halaman home dan sistem akan menampilkan direktori atau penyimpanan komputer pengguna. Kemudian pengguna memilih gambar bunga mawar yang ingin dicari. Selanjutnya, gambar akan di proses oleh sistem dengan mengekstraksi fitur warna menggunakan *color moment*, selanjutnya sistem akan membandingkan kemiripan gambar yang dicari dengan gambar yang ada di dalam *database* menggunakan *Euclidean distance*. Setelah itu, sistem menampilkan tampilan hasil pencarian.



Gambar 4.12 Sequence Diagram Login

Sequence diagram diatas menjelaskan pengguna yang ingin login. Pengguna dapat klik “login”, sistem menampilkan form login. Pengguna

memasukkan email dan *password*. Sistem kemudian melakukan validasi. Jika email dan *password* salah maka sistem akan menampilkan pesan bahwa email dan *password* salah. Jika benar, pengguna akan langsung masuk ke halaman *home* admin.

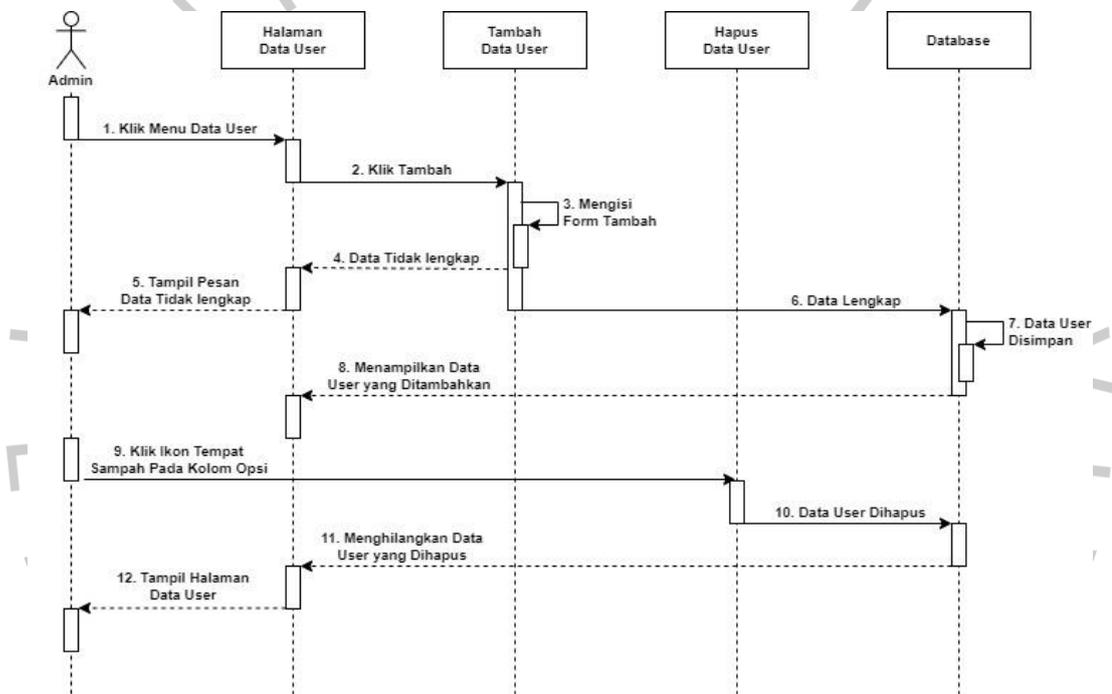


Gambar 4.13 Sequence Diagram Mengelola Data Gambar

Sequence diagram diatas menjelaskan pengguna yang ingin mengelola data gambar. Pengguna dapat melakukan kelola data gambar jika sudah melakukan *login*. Setelah melakukan *login*, pengguna dapat klik menu “data gambar” untuk masuk ke halaman data gambar. Berikut merupakan beberapa aktivitas yang dapat dilakukan pengguna untuk mengelola data gambar.

1. Pengguna dapat menambah data gambar dengan klik *button* “tambah”. Kemudian mengisi form. Jika data tidak lengkap, sistem akan menampilkan pesan bahwa data tidak lengkap. Jika data lengkap maka data gambar yang ditambahkan akan tersimpan di dalam *database* dan akan ditampilkan pada list halaman data gambar.

2. Pengguna dapat mengedit data gambar dengan klik “ikon pensil” di kolom opsi. Kemudian mengisi form edit. Jika data tidak lengkap, sistem akan menampilkan pesan bahwa data tidak lengkap. Jika data lengkap maka data gambar yang diperbarui akan tersimpan di dalam *database* dan akan ditampilkan pada list halaman data gambar.
3. Pengguna dapat menghapus data gambar dengan klik “ikon tempat sampah” pada kolom opsi. Setelah itu, sistem akan menghapus data gambar dari *database* dan list halaman data gambar.

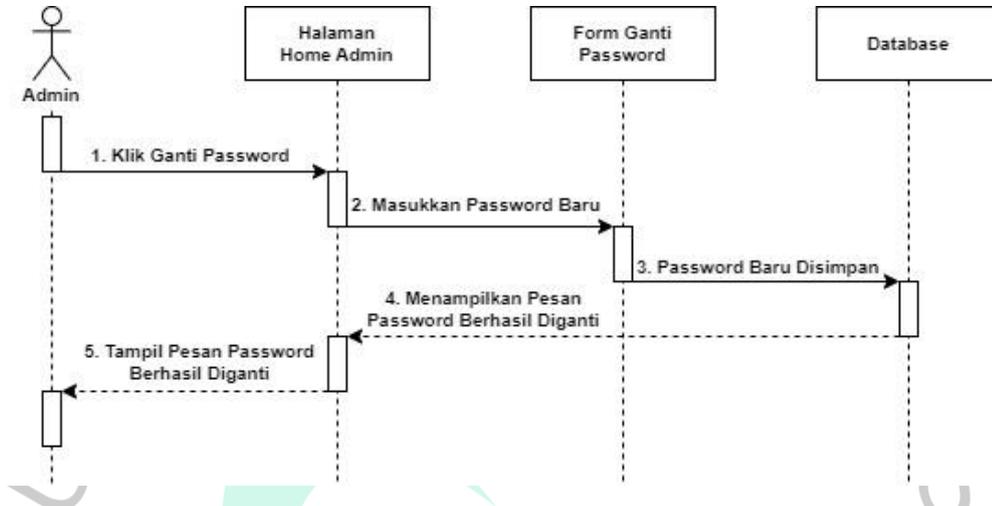


Gambar 4.14 Sequence Diagram Mengelola Data User

Sequence diagram diatas menjelaskan pengguna yang ingin mengelola data *user*. Pengguna dapat melakukan kelola data *user* jika sudah melakukan *login*. Setelah melakukan *login*, pengguna dapat klik menu “data *user*” untuk masuk ke halaman data *user*. Berikut merupakan beberapa aktivitas yang dapat dilakukan pengguna untuk mengelola data gambar.

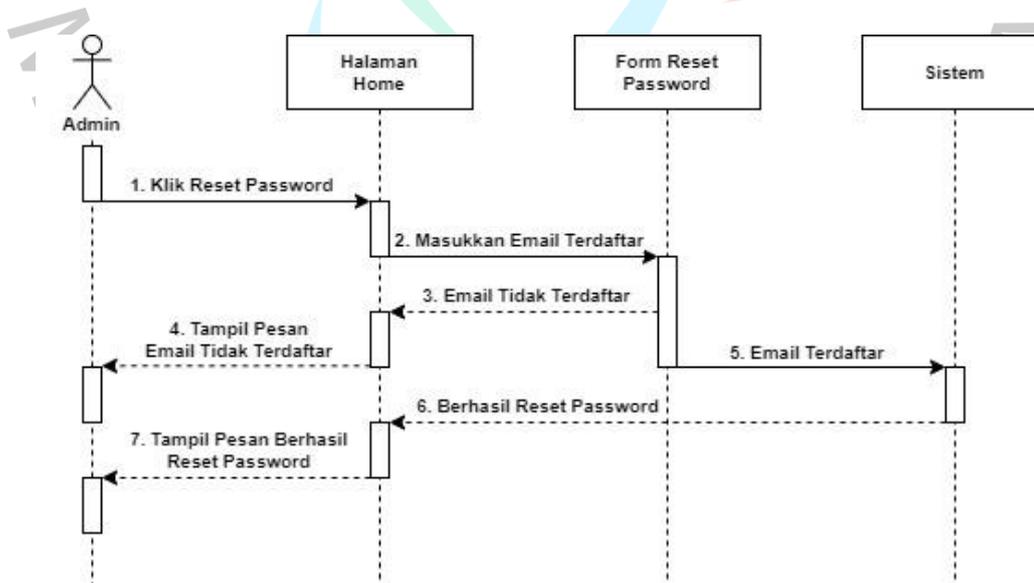
1. Pengguna dapat menambah data *user* dengan klik *button* “tambah”. Kemudian mengisi form. Jika data tidak lengkap, sistem akan menampilkan pesan bahwa data tidak lengkap. Jika data lengkap maka data *user* yang ditambahkan akan tersimpan di dalam *database* dan akan ditampilkan pada list halaman data *user*.

- Pengguna dapat menghapus data *user* dengan klik “ikon tempat sampah” pada kolom opsi. Setelah itu, sistem akan menghapus data *user* dari *database* dan list halaman data *user*.



Gambar 4.15 Sequence Diagram Mengganti Password

- Sequence diagram diatas menjelaskan pengguna yang ingin mengganti *password*. Pengguna dapat klik “ganti *password*”, selanjutnya mengisi form ganti *password* dengan *password* baru. Setelah itu, *password* baru akan disimpan ke dalam *database* dan sistem akan menampilkan pesan bahwa *password* berhasil diganti.



Gambar 4.16 Sequence Diagram Mereset Password

- Sequence diagram diatas menjelaskan pengguna yang ingin mereset *password*. Pengguna dapat klik “reset *password*” dan mengisi form dengan

memasukkan email yang terdaftar. Jika email tidak terdaftar maka sistem akan menampilkan pesan bahwa email tidak terdaftar. Jika email terdaftar maka sistem akan menampilkan bahwa *reset password* berhasil.

4.3.5 Perancangan Basis Data

Perancangan basis data merupakan proses rancangan dari sebuah basis data yang akan digunakan untuk menyimpan data dalam sebuah sistem. Pada aplikasi pencarian jenis bunga mawar, peneliti membuat basis data yang diperlukan untuk menyimpan dan mengolah data gambar dan data user. Berikut adalah rancangan basis data pada aplikasi pencarian jenis bunga mawar.

Tabel 4.10 Perancangan Basis Data Tabel Gambar

No	Field	Type	Length	Keterangan
1	gambarid(PK)	Int	5	id gambar
2	namagambar	Varchar	35	nama gambar
3	gambar	Varchar	100	gambar
4	mean	Double		nilai mean
5	deviation	Double		Nilai deviation
6	skewness	Double		nilai skewness
7	keterangan	Text		keterangan gambar

Tabel 4.11 Perancangan Basis Data Userlogin

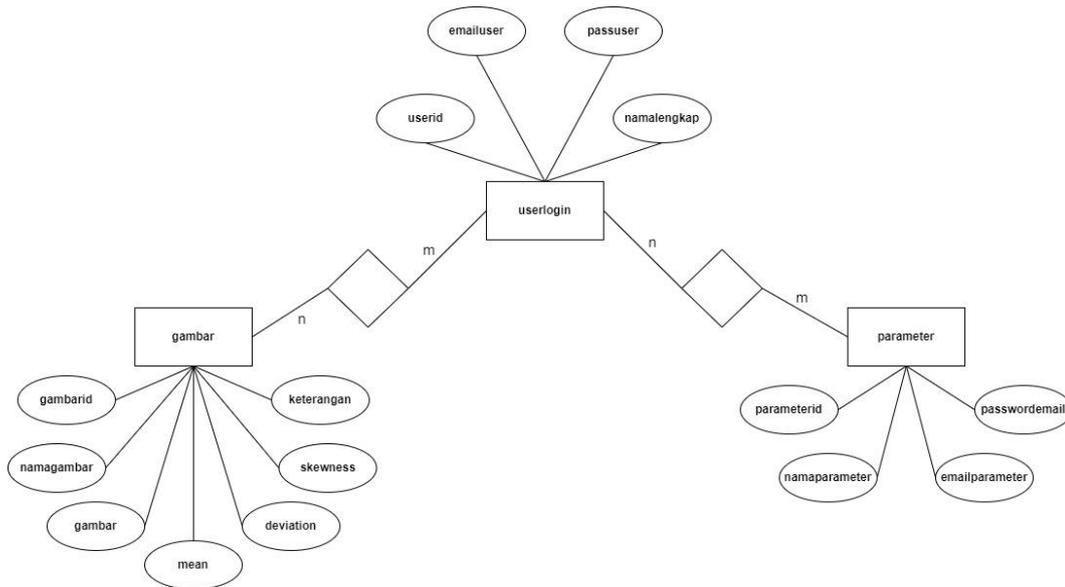
No	Field	Type	Length	Keterangan
1	userid(PK)	Int	5	id user
2	emailuser	Varchar	50	email user
3	passuser	Varchar	32	password user
4	namalengkap	Varchar	70	nama lengkap user

Tabel 4.12 Perancangan Basis Data Parameter

No	Field	Type	Length	Keterangan
1	parameterid(PK)	Int	2	id parameter
2	namaparameter	Varchar	50	nama parameter
3	emailparameter	Varchar	100	email parameter
4	passwordemail	Varchar	100	password email

4.3.6 Entity Relationship Diagram

Entity Relationship Diagram (ERD) merupakan model perancangan basis data yang menggambarkan hubungan antara entitas atau objek dalam sistem.

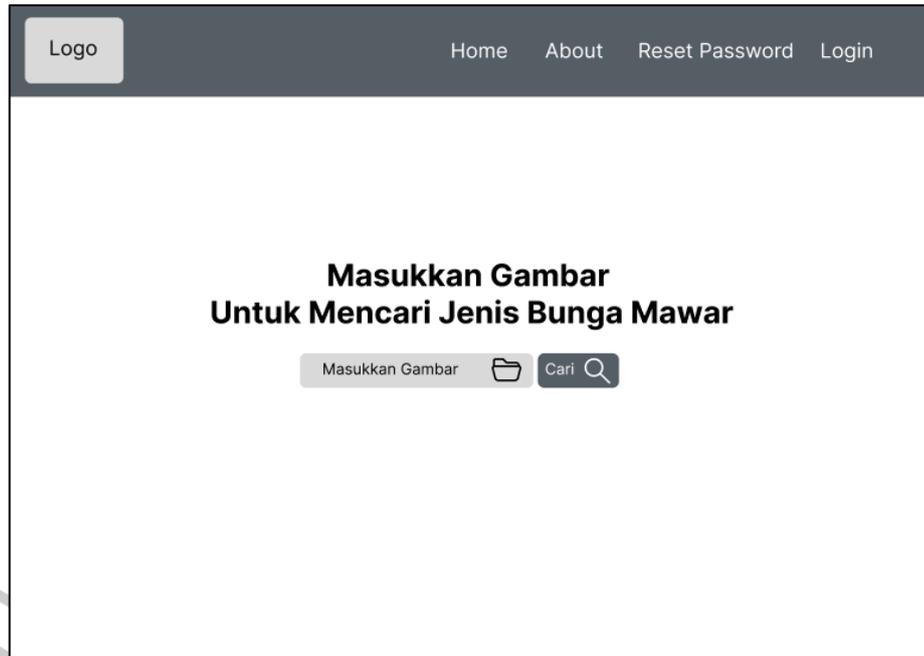


Gambar 4.17 Entity Relationship Diagram

Gambar di atas merupakan ERD diagram pada aplikasi pencarian jenis bunga mawar. Entitas parameter digunakan untuk mengelola berbagai atribut pada entitas userlogin. Entitas userlogin juga dapat mengelola berbagai atribut pada entitas gambar.

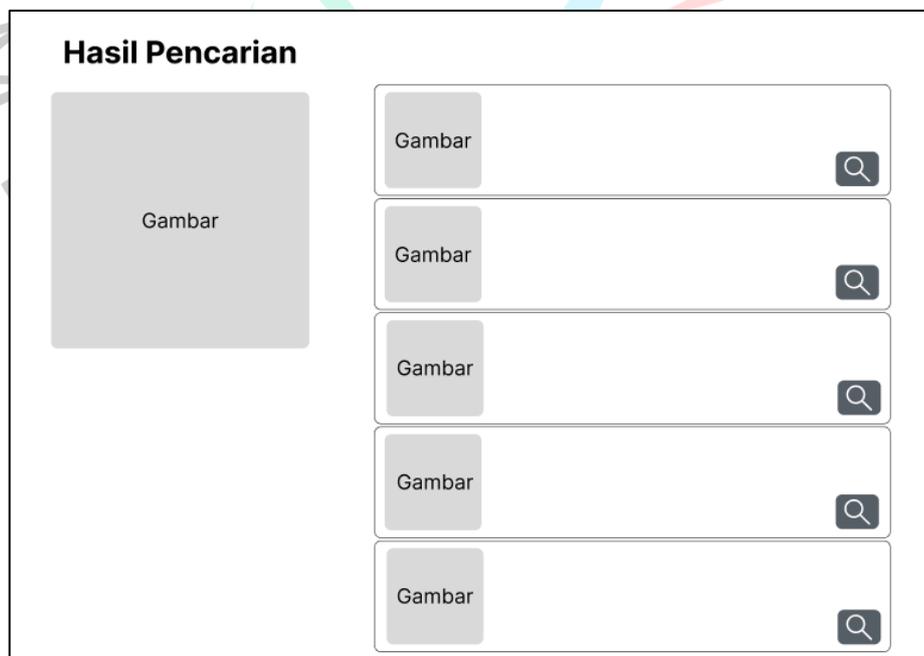
4.3.7 Perancangan Antarmuka

Perancangan antarmuka adalah proses yang bertujuan untuk merancang tampilan visual dan interaksi antara pengguna dan sistem. Selain itu, perancangan antarmuka juga membantu memudahkan programmer dalam mendesain aplikasi yang diinginkan. Berikut merupakan perancangan antarmuka aplikasi pencarian jenis bunga mawar.



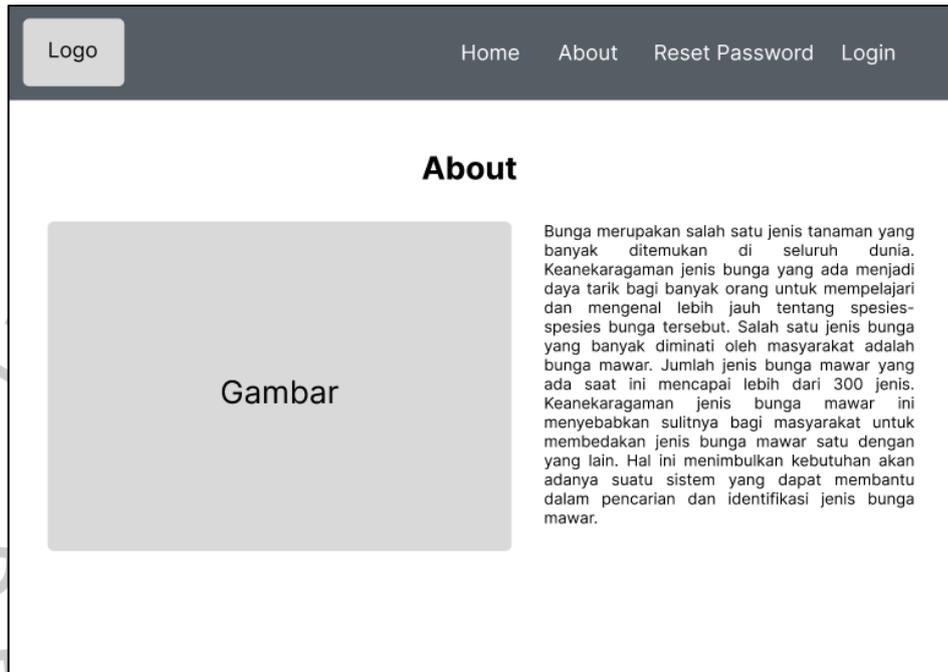
Gambar 4.18 Perancangan Antarmuka Halaman Home

Gambar diatas merupakan perancangan antarmuka untuk halaman home. Pada halaman home terdapat fitur utama pada aplikasi ini yaitu melakukan pencarian jenis bunga mawar dengan memasukkan gambar bunga mawar yang ingin dicari. Pada halaman tersebut juga terdapat beberapa menu yaitu *about*, *reset password*, *login*.



Gambar 4.19 Perancangan Antarmuka Hasil Pencarian

Gambar diatas merupakan perancangan antarmuka untuk tampilan hasil pencarian. Pada tampilan hasil pencarian akan menampilkan beberapa gambar yang dicari atau yang mirip dengan gambar yang dicari. Kemudian, terdapat ikon kaca pembesar yang direncanakan akan menampilkan keterangan dari gambar tersebut.



Gambar 4.20 Perancangan Antarmuka Halaman About

Gambar diatas merupakan perancangan antarmuka untuk halaman about. Pada halaman tersebut akan menampilkan gambar dan informasi tentang *website* tersebut.

The image shows a login form with a title 'Login'. It contains two input fields: 'Email' with the placeholder text 'Email User' and 'Password' with the placeholder text 'Password'. Below the input fields are two buttons: 'Login' and 'Batal'.

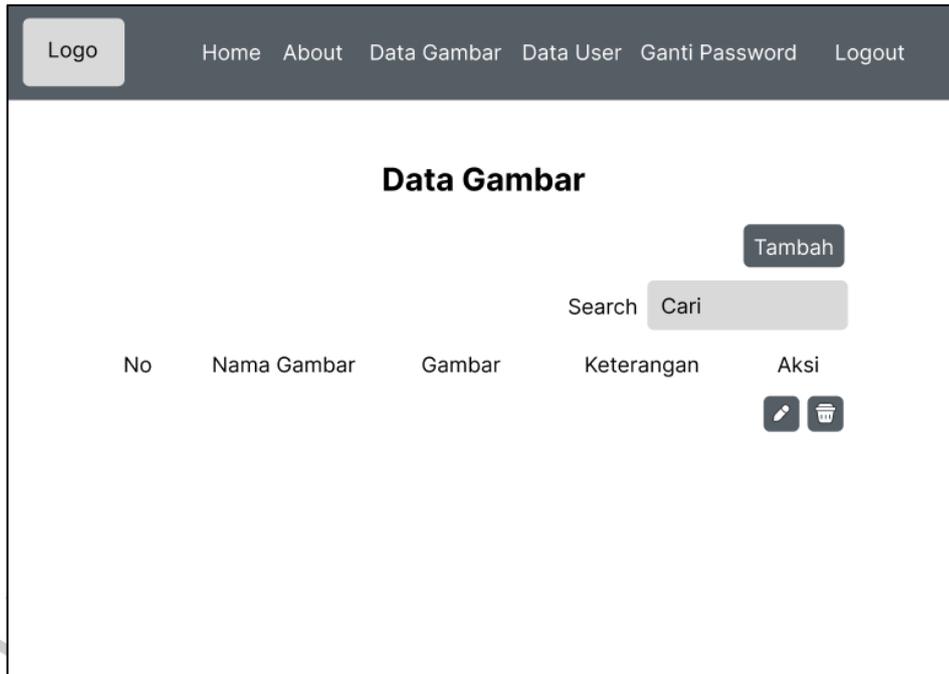
Gambar 4.21 Perancangan Antarmuka Form Login

Gambar diatas merupakan perancangan antarmuka *form login*. Pada *form* tersebut pengguna harus memasukkan email dan *password* yang terdaftar pada *website*.



Gambar 4.22 Perancangan Antarmuka Halaman Home Admin

Gambar diatas merupakan perancangan antarmuka untuk halaman home admin. Pada halaman home admin juga terdapat fitur utama pada aplikasi ini yaitu melakukan pencarian jenis bunga mawar dengan memasukkan gambar bunga mawar yang ingin dicari. Namun, perbedaannya adalah pada halaman tersebut terdapat beberapa menu tambahin seperti data gambar, data user dan ganti *password*.



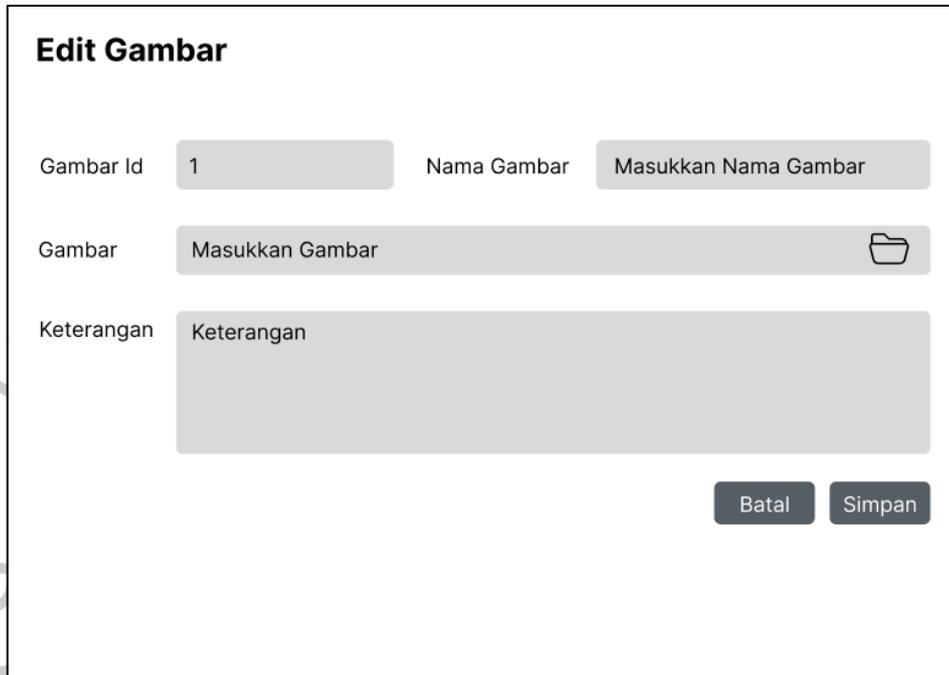
Gambar 4.23 Perancangan Antarmuka Halaman Data Gambar

Gambar diatas merupakan perancangan antarmuka untuk halaman data gambar. Pada halaman data gambar terdapat list yang menampilkan nomor, nama gambar, gambar, keterangan, dan aksi. List tersebut merupakan *dataset* yang telah dimasukkan pada *website*. Pengguna juga dapat mengelola data gambar dengan menambah, mengedit dan menghapus.



Gambar 4.24 Perancangan Antarmuka Tambah Data Gambar

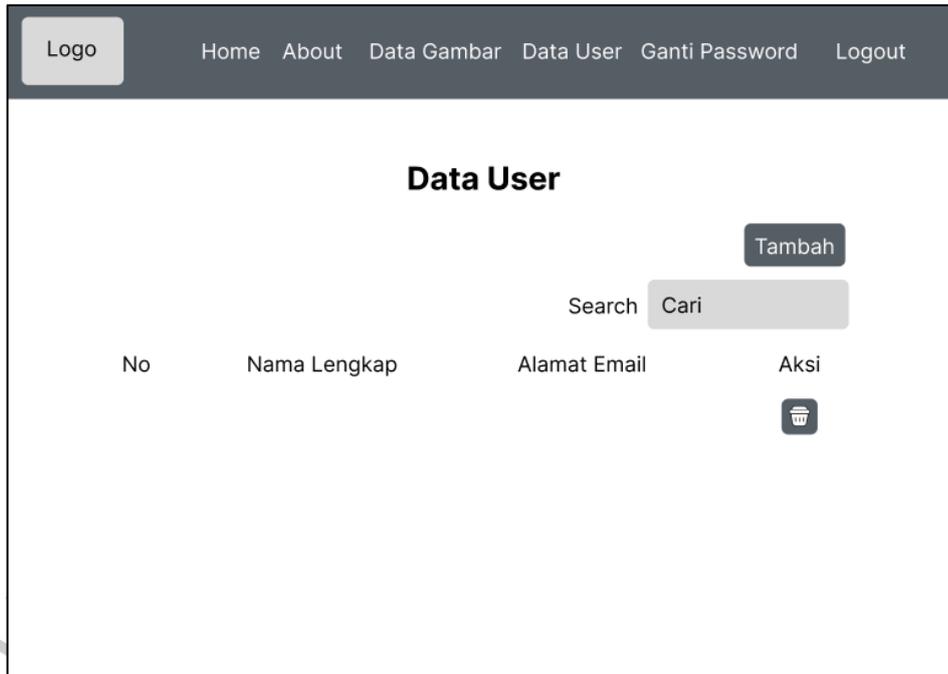
Gambar diatas merupakan perancangan antarmuka *form* tambah data gambar. Pada *form* tersebut pengguna harus memasukkan nama gambar, gambar dan keterangan. Data yang ditambah akan disimpan ke dalam *database*, kemudian ditampilkan pada list halaman data gambar.



The image shows a web form titled "Edit Gambar". It contains four input fields: "Gambar Id" with the value "1", "Nama Gambar" with the placeholder "Masukkan Nama Gambar", "Gambar" with the placeholder "Masukkan Gambar" and a folder icon, and "Keterangan" with the placeholder "Keterangan". At the bottom right, there are two buttons: "Batal" and "Simpan".

Gambar 4.25 Perancangan Antarmuka Edit Data Gambar

Gambar diatas merupakan perancangan antarmuka *form* edit data gambar. Pada *form* tersebut pengguna dapat memperbarui nama gambar, gambar dan keterangan. Data yang diperbarui akan disimpan ke dalam *database*, kemudian ditampilkan pada list halaman data gambar.



Gambar 4.26 Perancangan Antarmuka Halaman Data User

Gambar diatas merupakan perancangan antarmuka untuk halaman data *user*. Pada halaman data *user* terdapat list yang menampilkan nomor, nama lengkap, alamat email, dan aksi. List tersebut merupakan data *user* yang telah dimasukkan pada *website*. Pengguna juga dapat mengelola data *user* dengan menambah dan menghapus.

Gambar 4.27 Perancangan Antarmuka Form Tambah User

Gambar diatas merupakan perancangan antarmuka *form* tambah *user*. Pada *form* tersebut pengguna harus memasukkan nama lengkap dan email untuk menambahkan *user* baru.

4.3.8 Perancangan Pengujian

Perancangan pengujian adalah proses merancang dan mempersiapkan rencana pengujian untuk menguji perangkat lunak atau sistem secara menyeluruh. Perancangan pengujian pada penelitian ini menggunakan pengujian *black box* dan *white box*.

4.3.8.1 Perancangan Pengujian *Black Box*

Perancangan pengujian *black box* merupakan perancangan pengujian yang diterapkan pada semua fitur sebuah aplikasi untuk menentukan apakah output atau hasil yang diperoleh berkaitan dengan input yang diberikan. Berikut merupakan tabel skenario, input, dan hasil pengujian *black box*.

Tabel 4.13 Perancangan Pengujian *Black Box*

No	Skenario Pengujian	Input	Hasil Yang Diharapkan
1	Pencarian Gambar	Pengguna memasukan gambar bunga yang ingin dicari.	Berhasil menampilkan tampilan hasil pencarian gambar yang mirip dengan yang dicari.
2	Keterangan Gambar	Pengguna klik “ikon kaca pembesar” untuk melihat keterangan gambar yang diinginkan.	Berhasil menampilkan keterangan gambar.
3	<i>About</i>	Pengguna memilih menu <i>about</i> untuk masuk ke halaman <i>about</i> .	Berhasil masuk ke halaman <i>about</i> .
4	<i>Reset Password</i>	Pengguna memasukan email yang terdaftar untuk melakukan <i>reset password</i> .	Sistem berhasil mengirim <i>password</i> baru melalui email.
5	<i>Login</i>	Pengguna memasukkan email dan <i>password</i> untuk masuk ke <i>website</i> admin.	Berhasil masuk ke tampilan <i>website</i> admin.
6	Data Gambar	Pengguna memilih menu data gambar untuk masuk ke halaman data gambar.	Berhasil masuk ke halaman data gambar.
7	Tambah Data Gambar	Pengguna klik <i>button</i> “tambah” untuk menambah data gambar dengan mengisi <i>form</i> nama gambar, gambar dan keterangan.	Berhasil menambah data gambar.
8	Edit Data Gambar	Pengguna klik “ikon pensil” untuk mengedit data gambar dengan mengisi <i>form</i> nama gambar, gambar, dan keterangan.	Berhasil mengedit data gambar.
9	Hapus Data Gambar	Pengguna klik “ikon tempat sampah” untuk menghapus data gambar.	Berhasil menghapus data gambar.
10	Data User	Pengguna memilih menu data user untuk masuk ke halaman data user.	Berhasil masuk ke halaman data user.

11	Tambah Data User	Pengguna klik <i>button</i> “tambah” untuk menambah data user dengan mengisi <i>form</i> nama lengkap dan email.	Berhasil menambah data user.
12	Hapus Data User	Pengguna klik “ikon tempat sampah” untuk menghapus data user.	Berhasil menghapus data user.
13	Ganti <i>Password</i>	Pengguna memasukkan <i>password</i> baru.	Berhasil mengganti <i>password</i> .
14	<i>Logout</i>	Pengguna klik <i>button</i> “logout”.	Berhasil keluar dari tampilan <i>website</i> admin.

4.3.8.2 Perancangan Pengujian *White Box*

Perancangan pengujian *white box* merupakan perancangan untuk merancang aplikasi, karena dengan melakukan pengujian ini, dapat memastikan bahwa aplikasi berjalan dengan baik sesuai dengan skenario yang diinginkan.

Tabel 4.14 Perancangan Pengujian *White Box*

No	Hasil Yang Diharapkan	Source Code
1	Membuat gambar menjadi 1000 x 1000 piksel	<pre> // \$im_src = imagecreatefromjpeg(\$vfile_upload); \$src_width = imagesx(\$im_src1); \$src_height = imagesy(\$im_src1); \$dst_width = 1000; \$dst_height = 1000; \$im = imagecreatetruecolor(\$dst_width,\$dst_height); imagecopyresampled(\$im, \$im_src1, 0, 0, 0, 0, \$dst_width, \$dst_height, \$src_width, \$src_height); switch(\$img_t){ case 'image/jpeg': imagejpeg(\$im, \$vdir_upload.\$fupload_name); break; case 'image/png': imagepng(\$im, \$vdir_upload.\$fupload_name); break; } imagedestroy(\$im); </pre>
2	Menghitung nilai Mean	<pre> // Menghitung nilai Mean for (\$i = 0; \$i < \$width; \$i++) { for (\$j = 0; \$j < \$height; \$j++) { \$pixel = imagecolorat(\$image, \$i, \$j); \$grayValue = (\$pixel >> 8) & 0xFF; \$mean += \$grayValue; } } \$mean /= (\$width * \$height); </pre>

3	Menghitung nilai Standard Deviation dan Skewness	<pre>// Menghitung nilai Standard Deviation dan Skewness for (\$i = 0; \$i < \$width; \$i++) { for (\$j = 0; \$j < \$height; \$j++) { \$pixel = imagecolorat(\$image, \$i, \$j); \$grayValue = (\$pixel >> 8) & 0xFF; \$standardDeviation += pow(\$grayValue - \$mean, 2); \$skewness += pow(\$grayValue - \$mean, 3); } } \$standardDeviation = sqrt(\$standardDeviation / (\$width * \$height)); \$skewness /= pow(\$standardDeviation, 3) * (\$width * \$height);</pre>
4	Menghitung Euclidean Distance	<pre>while (\$image = mysqli_fetch_array(\$result)) { \$gambarrgb = \$image['gambar']; \$mean1 = (float) \$image['mean']; \$deviation1 = (float) \$image['deviation']; \$skewness1 = (float) \$image['skewness']; \$meanr = pow((\$mean - \$mean1), 2); \$deviationr = pow((\$deviation - \$deviation1), 2); \$skewnessr = pow((\$skewness - \$skewness1), 2); \$test = \$meanr + \$deviationr + \$skewnessr; \$d = sqrt(\$meanr + \$deviationr + \$skewnessr); \$scores[\$image['gambarid']] = array("id" => \$image['gambarid'], "distance" => \$d, "gambar" => \$image['gambar'], "namagambar" => \$image['namagambar'], "keterangan" => \$image['keterangan']); }</pre>

