

BAB III

METODOLOGI PENELITIAN

3.1 Subjek dan Objek Penelitian

3.1.1 Subjek Penelitian

Subjek penelitian adalah objek, fenomena, atau individu yang digunakan untuk mengumpulkan data atau informasi tentang variabel penelitian. Subjek penelitian yang menjadi sasaran peneliti adalah di Klinik MCM Beauty Center Ajibarang.

3.1.2 Objek Penelitian

Objek penelitian yaitu titik pusat perhatian dalam suatu penelitian, dalam penelitian ini adalah data gejala-gejala yang dapat mengidentifikasi masalah kulit wajah yang dapat memberikan ciri-ciri dan saran *treatment* sesuai studi kasus.

3.2 Alat dan Bahan Penelitian

Dalam penelitian ini digunakan alat dan bahan sebagai penunjang keberhasilan penelitian. Adapun alat dan bahan yang dimaksud yaitu:

3.2.1 Alat

Alat yang digunakan dalam penelitian ini terdiri dari dua jenis, yaitu perangkat keras (*hardware*) dan perangkat lunak (*software*):

1. Perangkat Keras (*Hardware*)

Device : DESKTOP-TJ8OUVE

Processor : Intel(R) Core(TM) i3-6006U CPU @
2.00GHz 1.99 GH

RAM : 8 GB

2. Perangkat Lunak (*Software*)

Sistem Operasi : *Windows 10-Pro 64 bit*

Bahasa Pemrograman: *PHP framework Laravel, framework CSS*

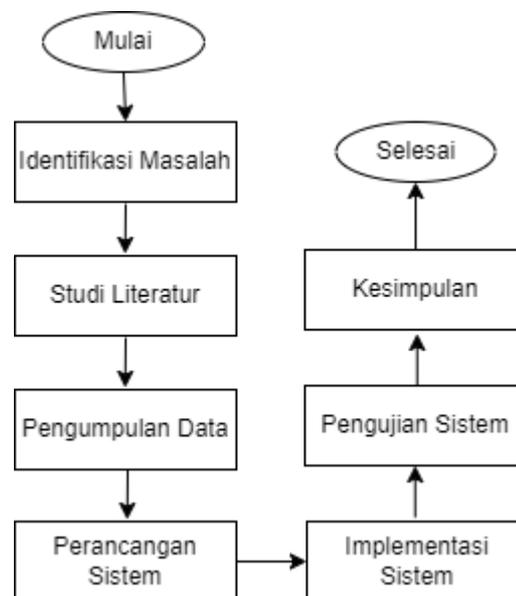
Bootstrap

Aplikasi : *Visual Studio Code, XAMPP V3.3.0, MySQL, Web Browser, Microsoft Office (Word, Powerpoint, Excel), Balsamiq Wireframe.*

3.2.2 Bahan

Studi literatur, data gejala masalah kulit wajah, solusi *treatment* kulit wajah, wawancara terhadap pakar, dan observasi ahli merupakan sumber informasi yang digunakan dalam penelitian ini

3.3 Diagram Alir Penelitian



Gambar 3. 1 Diagram Alir Penelitian

Laporan ini disusun melalui beberapa tahapan. Diagram alir metodologi penelitian pada Gambar 3.1 menggambarkan langkah-langkah yang meliputi: identifikasi masalah, studi literatur, pengumpulan data, perancangan sistem, implementasi sistem, pengujian, serta kesimpulan.

3.3.1 Identifikasi Masalah

Tahap pertama dalam penelitian adalah proses identifikasi masalah, yang didasarkan pada konteks spesifik dari permasalahan yang dihadapi. Dalam hal ini, pasien sering mengalami keterbatasan pengetahuan mengenai masalah kulit wajah seperti jerawat, melasma, dan penuaan, yang mengakibatkan kesalahan dalam memilih perawatan. Selain itu, keterbatasan waktu dan kesulitan dalam menjadwalkan janji temu dengan dokter estetika memperburuk situasi. Oleh karena itu, diperlukan solusi inovatif yang dapat menyediakan akses cepat dan mudah secara online terhadap informasi kesehatan kulit wajah beserta rekomendasi perawatan yang tepat.

3.3.2 Studi Literatur

Referensi penelitian sebelumnya dan teori yang sebanding ditemukan dalam jurnal, makalah, artikel *online*, dan sumber lainnya digunakan dalam studi literatur untuk mendapatkan informasi. Untuk memahami permasalahan yang diangkat dalam penelitian ini, yang mencakup berbagai masalah kulit wajah, gejala, dan metode perawatan untuk mengatasi masalah kulit wajah, dilakukan penyelidikan berbagai sumber. Selain itu, mencari informasi tentang metode *case based reasoning* dan algoritma *nearest neighbor*.

3.3.3 Pengumpulan Data

Pada tahap ini penulis mengumpulkan informasi dengan melakukan interaksi langsung melalui wawancara kepada pakar dr. Bergas Widadi, seorang dokter estetika di Klinik MCM Beauty Center Ajibarang untuk mendapatkan data yang dibutuhkan, memperkuat data dan mempelajari masalah kulit wajah, termasuk bobot nilai pakar, berbagai masalah dan gejala kulit wajah, dan data korelasi masalah kulit wajah dengan gejala masalah pada kulit wajah.

Tabel 3. 1 Nilai Bobot Pakar[5]

Nilai bobot pakar	Keterangan
5	Gejala Penting
3	Gejala Sedang
1	Gejala Biasa

Tabel 3.1 merupakan hasil yang diperoleh dari wawancara untuk menentukan nilai bobot pada metode *case based reasoning*.

Tabel 3. 2 Masalah Kulit Wajah

Jenis Masalah kulit wajah
Jerawat (<i>Acne</i>)
Melasma (Flek)
Penuaan (<i>Aging</i>)

Tabel 3.2 menunjukkan jenis masalah kulit wajah, yang terdiri atas tiga macam yakni jerawat (*acne*), melasma (flek), penuaan (*aging*). Dataset gejala dapat diuraikan pada Tabel 3.3 berikut.

Tabel 3. 3 Gejala Masalah Kulit Wajah

Kode Gejala	Gejala
G001	Munculnya garis-garis halus dan kerutan pada kulit wajah
G002	Kulit cenderung berminyak
G003	Terjadinya peningkatan melanin di kulit sehingga lebih gelap dibanding kulit sekitar
G004	Muncul benjolan kecil dan meradang
G005	Kulit cenderung kehilangan elastisitasnya
G006	Muncul benjolan besar, keras, dan meradang yang berkembang di bawah kulit
G007	Terjadi tanda-tanda peradangan pada kulit
G008	Adanya bercak-bercak gelap pada kulit wajah, terutama di area seperti pipi, dahi, dan daerah sekitar bibir
G009	Pori-pori besar
G010	Muncul secara simetris di kedua sisi wajah, artinya bercak-bercak gelap muncul pada kedua sisi wajah dengan pola yang mirip
G011	Bercak-bercak melasma dapat bertambah besar seiring berjalannya waktu

Kode Gejala	Gejala
G012	Penurunan produksi minyak alami kulit
G013	Muncul benjolan berisi nanah yang dapat muncul pada kulit
G014	Munculnya komedo
G015	Pigmentasi (warna) kulit yang tidak merata
G016	Produksi kolagen dan elastin kulit menurun

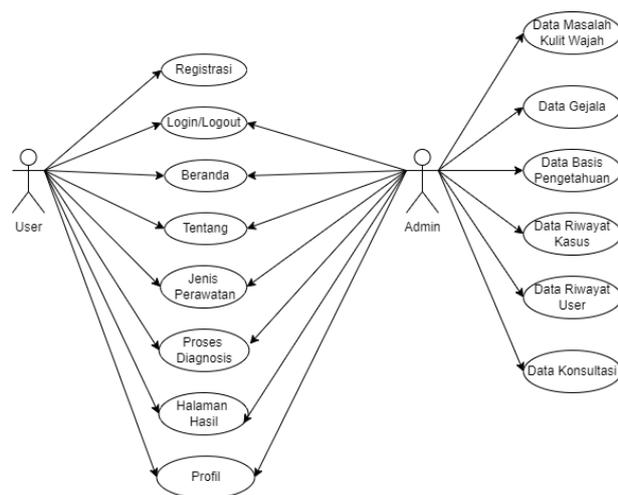
Tabel 3.3 menampilkan gejala beserta kode gejala dari wawancara pakar langsung.

3.3.4 Perancangan Sistem

Alur sistem dijelaskan melalui *Diagram Unified Modelling Language (UML)*, yang terdiri dari *Use Case Diagram*, *Activity Diagram*, *Class Diagram* dan *Sequence Diagram*.

3.3.4.1 Use Case Diagram

Use Case Diagram adalah desain sistem yang menggambarkan interaksi *actor* dengan sistem yang akan dikembangkan. Diagram ini sangat penting sebagai panduan selama proses pengembangan sistem, karena menjelaskan interaksi antara *actor* dan sistem. Gambar 3.2 menunjukkan *use case diagram* dan secara singkat menjelaskan hubungan antara *use case*, *actor*, dan sistem.



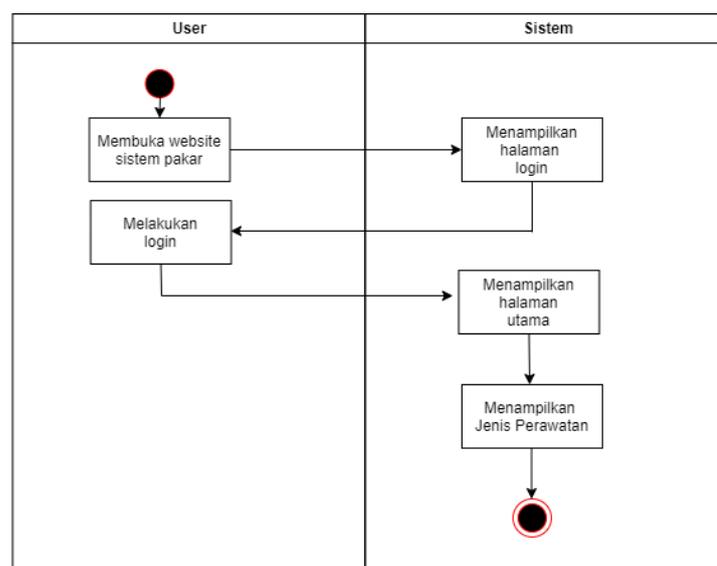
Gambar 3. 2 Use Case Diagram

Berdasarkan Gambar 3. 2 diatas dapat dipaparkan bahwa dalam sistem pakar ini memiliki dua *actor* yaitu pengguna (*user*) dan administrator (*admin*).

Pertama, sebelum dapat mengakses sistem, *user* harus melakukan registrasi dengan memasukkan data pribadi mereka. Setelah berhasil registrasi maka *user* dapat *login* ke sistem dan melihat tampilan beranda mengenai tentang kami, informasi masalah kulit wajah, dan tujuan dibuat *website* tersebut. Kemudian ketika akan melakukan diagnosis gejala yang dialami, *user* harus menandai gejala-gejala yang muncul. Tampilan akhir yang akan muncul adalah sebuah tampilan hasil diagnosis. Apabila *user* ingin mengedit profil bisa klik menu profil di sebelah paling kanan. Kedua, jika *user* adalah admin maka cukup *login* tanpa melalui registrasi untuk masuk ke sistem. Di dalam sistem, admin dapat mengelola data, seperti data masalah kulit wajah, data gejala, data basis pengetahuan, data riwayat kasus, data riwayat *user* dan menu *revise* pakar berupa data konsultasi. Prosesnya berupa CRUD yaitu menambah, mengedit dan menghapus kasus. Setelah selesai melakukan diagnosis baik *user* maupun admin dapat melakukan *logout* akun.

3.3.4.2 Activity Diagram

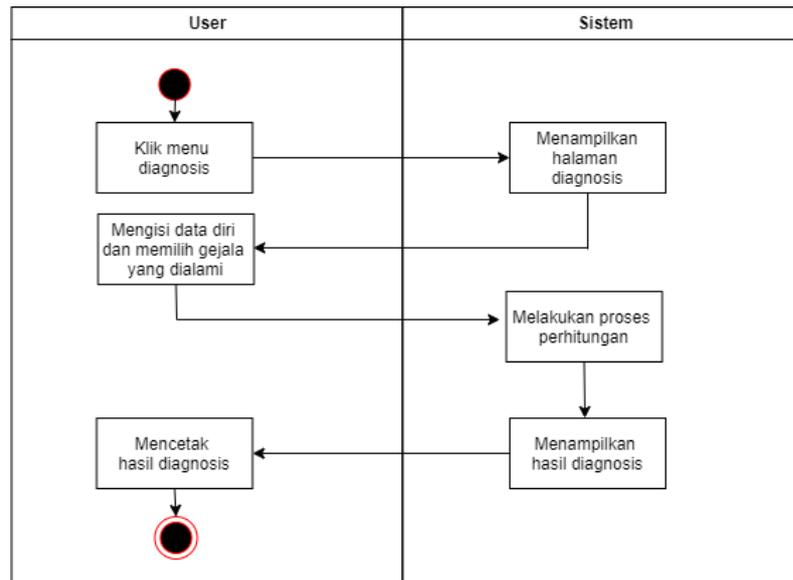
3.3.4.2.1 Activity Diagram Login User



Gambar 3.3 Activity Diagram Login User

Gambar 3.3 menunjukkan aktivitas diagram *login* dari sisi *user* yang dimulai dengan membuka *website* sistem pakar. Setelah itu, jika *user* memiliki akun, mereka diarahkan ke halaman utama dan dapat melihat halaman jenis perawatan.

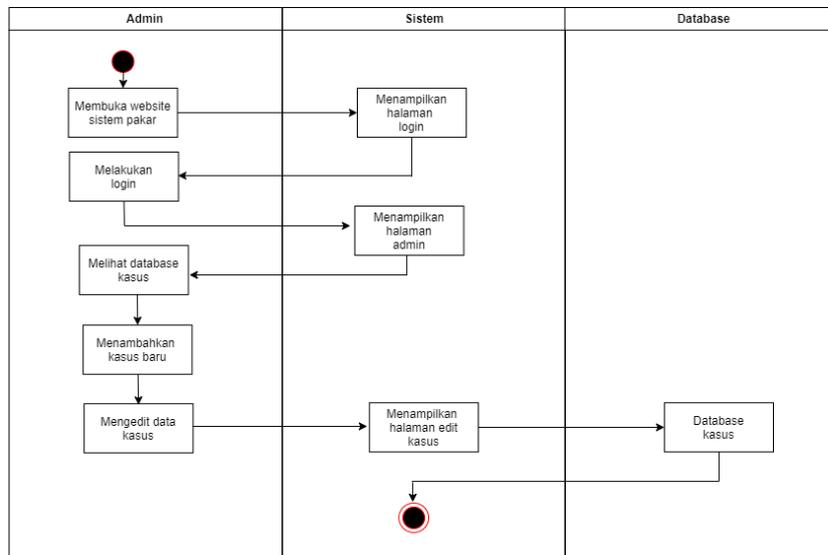
3.3.4.2.2 Activity Diagram Diagnosis User



Gambar 3.4 Activity Diagram Diagnosis User

Diagram aktivitas diagnosis pengguna dapat dilihat pada Gambar 3.4. Untuk memulai, pengguna perlu mengklik menu *navbar* diagnosis. Selanjutnya, sistem akan menampilkan halaman diagnosis. Pada halaman ini, pengguna dapat mengisi data diri mereka dan mencentang gejala yang dialami (minimal 4 gejala). Setelah itu, sistem akan melakukan proses perhitungan menggunakan algoritma *Nearest Neighbor*. Pada tahap perhitungan ini, sistem akan menerapkan metode *Case Based Reasoning* dengan mencari kasus-kasus yang paling mirip untuk menghasilkan kesimpulan dan rekomendasi perawatan. Hasil diagnosis ditampilkan oleh sistem yang kemudian dapat dicetak hasil perhitungannya oleh *user*.

3.3.4.2.3 Activity Diagram Admin

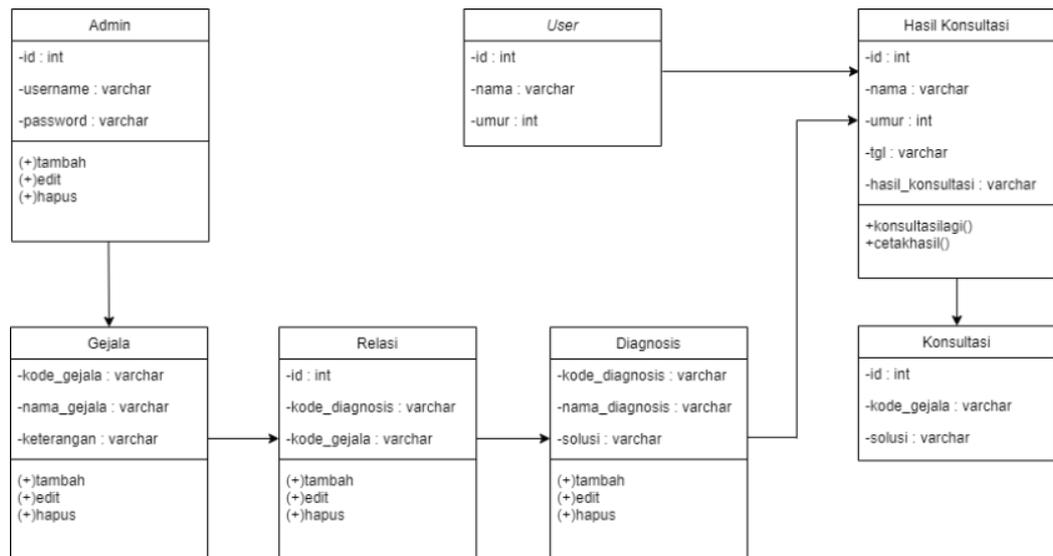


Gambar 3. 5 Activity Diagram Admin

Proses aktivitas dari sisi admin dimulai dengan membuka *website* sistem pakar dan login, seperti yang ditunjukkan pada gambar 3.5. Data kasus ditampilkan pada halaman admin selanjutnya. Ada dua menu, yang pertama memungkinkan admin menambah data kasus baru dengan mengisi form, dan yang kedua memungkinkan admin mengedit data kasus yang sudah ada pada database kasus.

3.3.4.3 Class Diagram

Class Diagram menggambarkan hubungan antara kelas-kelas dalam sistem yang dibangun dan cara kerja sama mereka untuk mencapai tujuan. Gambar 3.6 menampilkan diagram kelas dari sistem identifikasi masalah kulit wajah.



Gambar 3. 6 *Class Diagram*

Dapat diperhatikan pada gambar 3.6, *class diagram* yang merinci struktur dan interaksi antar kelas yang terlibat dalam sistem. Terdapat 7 kelas yang saling berhubungan untuk menggambarkan *user*.

1. Kelas Admin

Kelas Admin pada gambar mempresentasikan pengelola sistem. Kelas admin memiliki atribut id, username, dan password.

2. Kelas User

Kelas User ini mempresentasikan pengguna sistem. Kelas user memiliki atribut id, nama, umur.

3. Kelas Gejala

Kelas Gejala merepresentasikan jenis gejala yang memiliki atribut kode_gejala, nama_gejala, keterangan.

4. Kelas Relasi

Kelas Relasi merepresentasikan relasi antara gejala dan diagnosis, dimana memiliki atribut id, kode_diagnosis, kode_gejala.

5. Kelas Diagnosis

Kelas Diagnosis merepresentasikan berupa atribut kode diagnosis, nama_diagnosis, dan solusi.

6. Kelas Konsultasi

Kelas Konsultasi merepresentasikan id, kode_gejala, solusi.

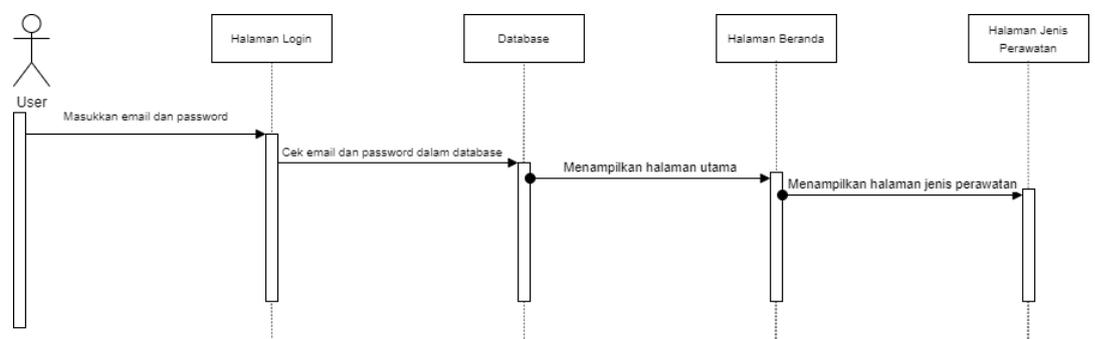
7. Kelas Hasil Konsultasi

Kelas Hasil Konsultasi merepresentasikan id, nama, umur, tgl, dan hasil_konsultasi.

3.3.4.4 *Sequence Diagram*

Sequence Diagram sebagai visualisasi yang menggambarkan interaksi antara objek atau entitas dalam sistem secara berurutan. *Sequence Diagram* membantu dalam memahami bagaimana objek atau komponen saling berinteraksi satu sama lain dalam suatu skenario. *Sequence Diagram* ini mengilustrasikan urutan pesan atau panggilan fungsi antar objek dalam sistem pakar. Setiap entitas direpresentasikan oleh garis vertikal yang disebut "*lifeline*" dan pesan-pesan antar entitas diurutkan secara temporal dari atas ke bawah.

3.3.4.4.1 *Sequence Diagram User - Login*

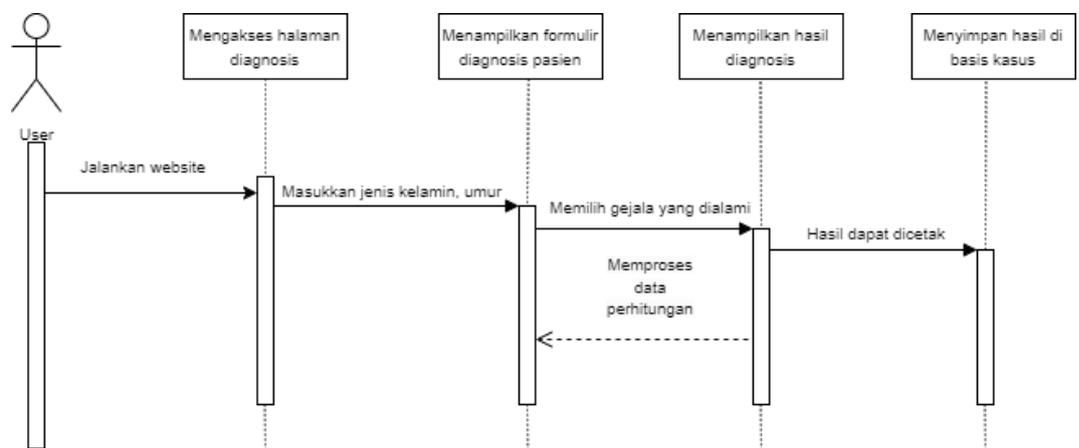


Gambar 3. 7 *Sequence Diagram User - Login*

Pada gambar 3.7 menampilkan proses dimulai ketika pengguna (user) memasukkan kredensial login mereka, yaitu email dan password, ke dalam

antarmuka login. Setelah pengguna mengirimkan data tersebut, antarmuka login meneruskan kredensial tersebut ke layanan otentikasi. Layanan otentikasi kemudian memeriksa validitas kredensial dengan mencocokkan username dan password yang diberikan dengan data yang ada di database pengguna. Selanjutnya pengguna dapat mengakses halaman beranda, dan halaman jenis perawatan.

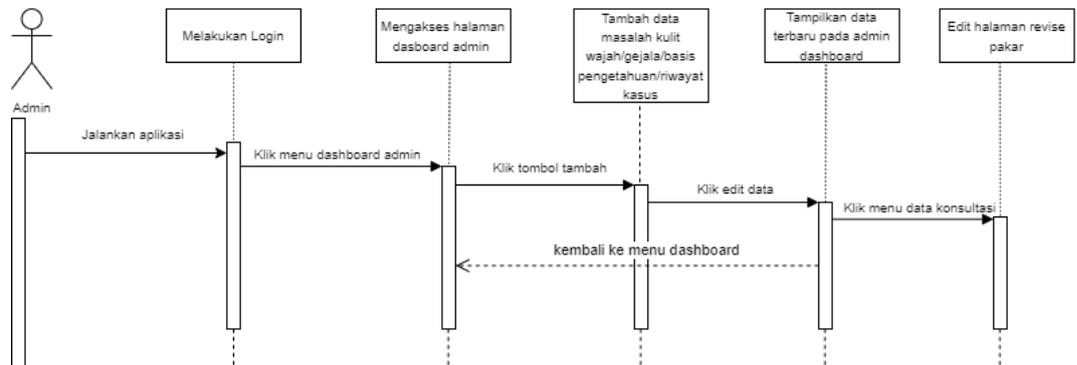
3.3.4.4.2 Sequence Diagram User – Diagnosis



Gambar 3. 8 Sequence Diagram User – Diagnosis

Pada gambar 3.8 menampilkan proses ketika user mengakses menu diagnosis, lalu memasukkan jenis kelamin dan umur. Pada menu diagnosis akan ditampilkan formulir diagnosis untuk user dapat memilih gejala yang dirasakan dan sistem akan melakukan proses perhitungan untuk menampilkan hasil diagnosis berupa *similarity* terbesar dari kasus yang ada, lalu terdapat fitur cetak data untuk melakukan penyimpanan terkait data hasil diagnosis agar dapat diunduh di perangkat. Selanjutnya hasil diagnosis akan disimpan sebagai basis kasus baru.

3.3.4.2.4 Sequence Diagram Admin – Dashboard



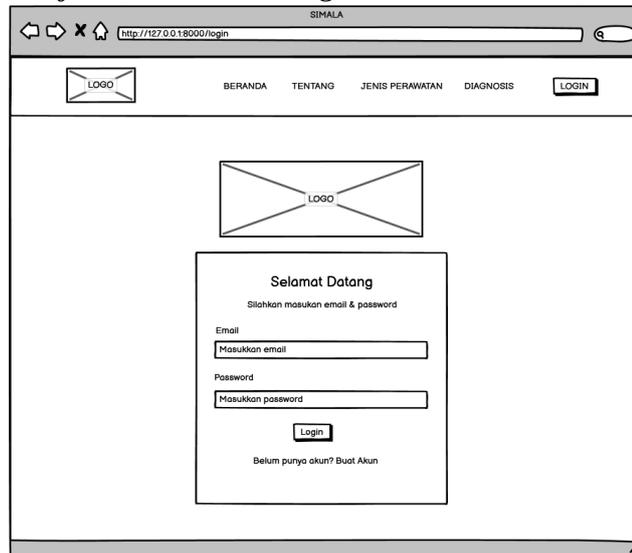
Gambar 3. 9 Sequence Diagram Admin – Dashboard

Pada gambar 3.9 menampilkan proses pada halaman dashboard yaitu dimulai dari admin melakukan login sebelum bisa mengakses dashboard, setelah masuk admin dapat melakukan tambah dan edit data pada menu data masalah kulit wajah, data gejala, data basis pengetahuan, data riwayat kasus. Lalu sistem akan menampilkan data terbaru yang telah diedit oleh admin, pada menu dashboard juga ada *revise* pakar yang berisi data konsultasi ketika ada kasus yang masuk membutuhkan solusi dari pakar (nilai *similarity* yang diberikan $< threshold$ yang ditentukan).

3.3.5 Perancangan Halaman Wireframe Sistem

Penulis merancang desain sistem (*wireframe*) web sistem pakar masalah kulit wajah sebelum mengimplementasikan kode bertujuan untuk mengembangkan antarmuka pengguna (UI). Adapun susunan rancangan sistem yang telah penulis buat sebagai berikut.

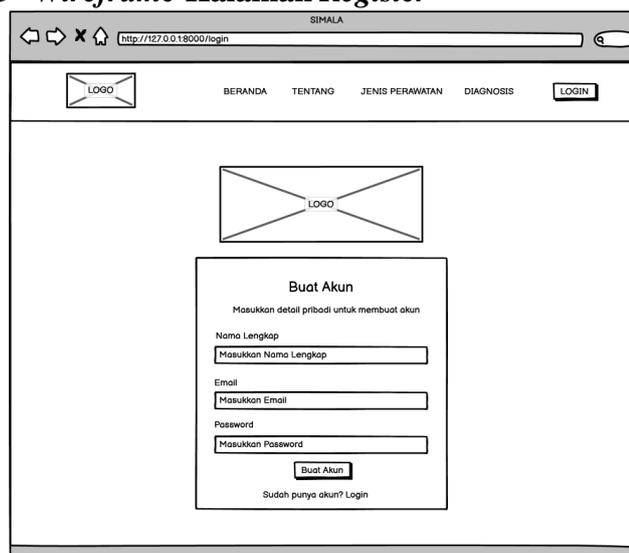
3.3.5.2 Wireframe Halaman Login



Gambar 3. 10 Wireframe Halaman Login

Pada gambar 3.10 menampilkan *wireframe* untuk halaman *login* dimana terdapat *input* berupa email dan password yang wajib dimasukan oleh pengguna sebelum masuk ke dalam proses diagnosis di sistem *website* yang dibuat. Setelah data diisi pengguna bisa klik *login*, sementara di bawah *button login* terdapat link apabila *user* belum memiliki akun yang mengarah ke halaman *register* akun.

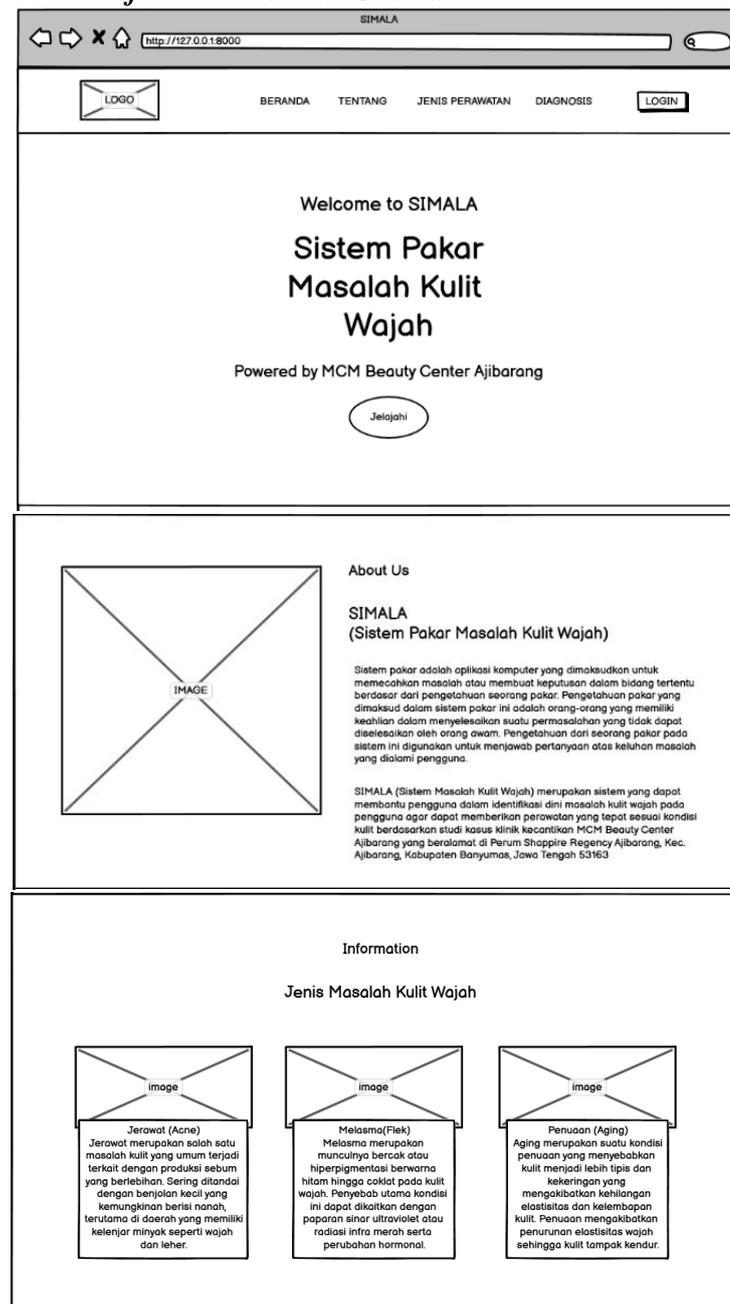
3.3.5.3 Wireframe Halaman Register

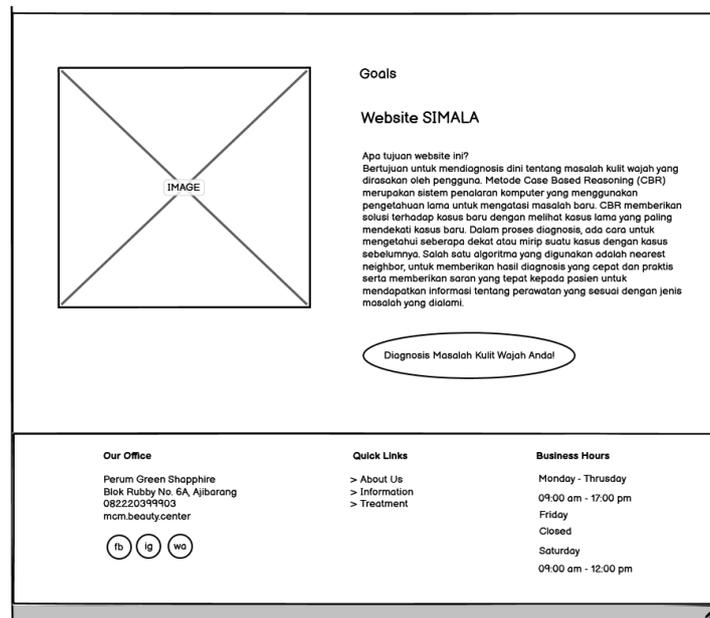


Gambar 3. 11 Wireframe Halaman Register

Gambar 3.11 merupakan *wireframe* halaman *register*, jadi sebelum pengguna masuk ke *website* sistem pakar untuk proses diagnosis harus mendaftarkan diri berupa nama lengkap, email dan password. Pada laman ini apabila pengguna sudah memiliki akun bisa klik link sudah punya akun yang akan diarahkan ke halaman *login*.

3.3.5.4 Wireframe Halaman Utama

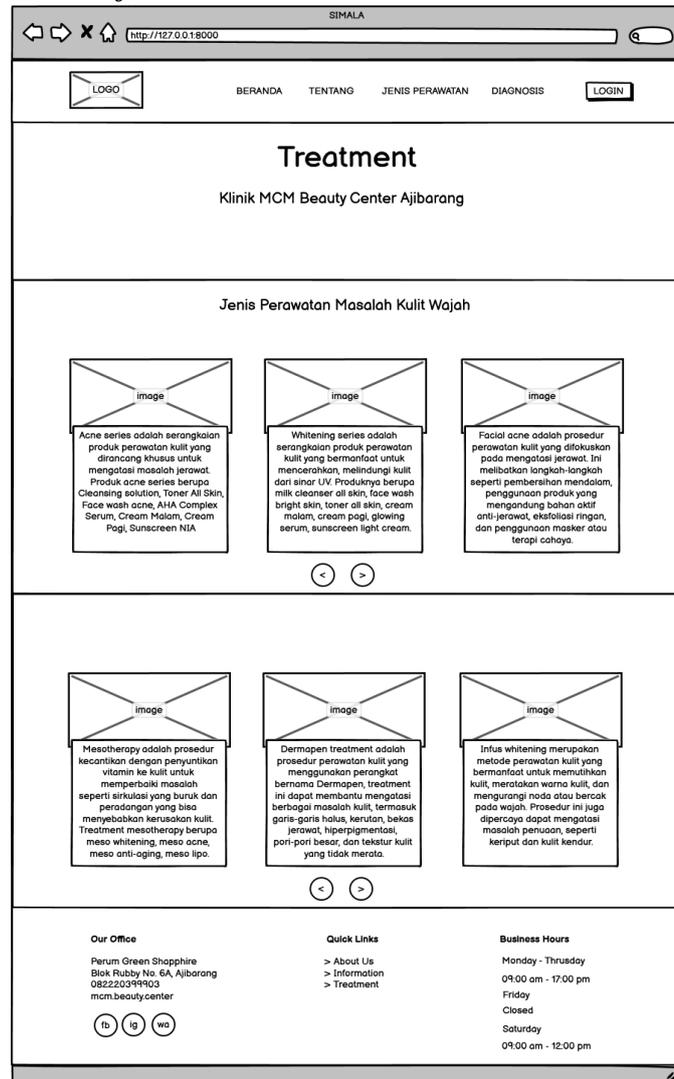




Gambar 3. 12 Wireframe Halaman Utama

Gambar 3.12 ditampilkan menu beranda ketika pengguna membuka *website* Sistem Pakar Masalah Kulit Wajah (SIMALA). Adapun isi dari halaman ini berupa logo, *navbar* yang memuat beranda, tentang, jenis perawatan dan diagnosis. Di halaman ini kita dapat menjelajahi mengenai apa itu sistem pakar, informasi jenis masalah kulit wajah dan tujuan dibuatnya *website* ini, serta terdapat *footer* yang memuat informasi alamat klinik, jam buka, dan *quick links*.

3.3.5.5 Wireframe Halaman Jenis Perawatan



Gambar 3. 13 Wireframe Halaman Jenis Perawatan

Pada gambar 3.13 *user* dapat memperoleh informasi tentang *treatment* yang disediakan di klinik MCM Beauty Center Ajibarang, dengan tampilan berupa *carousel*, *user* dapat melakukan *scroll* ke kanan dan kiri untuk melihat jenis perawatannya.

3.3.5.6 Wireframe Halaman Diagnosis

SIMALA

http://127.0.0.1:8000/diagnosa

LOGO BERANDA TENTANG JENIS PERAWATAN DIAGNOSIS LOGIN

Diagnosis Masalah Kulit Anda!

Kenali jenis kulit untuk treatment yang sesuai

FORMULIR DATA PENGGUNA

Nama Lengkap

Jenis Kelamin

Umur Tahun

No.	Kode Gejala	Nama Gejala	
1	G001	Munculnya garis-garis halus dan kerutan pada kulit wajah	<input type="checkbox"/>
2	G002	Kulit cenderung berminyak	<input type="checkbox"/>
3	G003	Terjadinya peningkatan melanin di kulit sehingga lebih gelap dibanding kulit sekitar	<input type="checkbox"/>
4	G004	Muncul benjolan kecil dan meradang	<input type="checkbox"/>
5	G005	Kulit cenderung kehilangan elastisitasnya	<input type="checkbox"/>
6	G006	Muncul benjolan besar keras dan meradang yang berkembang di bawah kulit	<input type="checkbox"/>
7	G007	Terjadi tanda-tanda peradangan pada kulit	<input type="checkbox"/>
8	G008	Adanya bercak-bercak gelap pada kulit wajah terutama di area seperti pipi dahi dan daerah sekitar bibir	<input type="checkbox"/>
9	G009	Pori-pori besar	<input type="checkbox"/>
10	G010	Muncul secara simetris di kedua sisi wajah artinya bercak-bercak gelap muncul pada kedua sisi wajah dengan pola yang mirip	<input type="checkbox"/>
11	G011	Bercak-bercak melasma dapat bertambah besar seiring berjalannya waktu	<input type="checkbox"/>
12	G012	Penurunan produksi minyak alami kulit	<input type="checkbox"/>
13	G013	Muncul benjolan berisi nanah yang dapat muncul pada kulit	<input type="checkbox"/>
14	G014	Munculnya komedo	<input type="checkbox"/>
15	G015	Pigmentasi (warna) kulit yang tidak merata	<input type="checkbox"/>
16	G016	Produksi kolagen dan elastin kulit menurun	<input type="checkbox"/>

Our Office
Perum Green Shoppire
Blok Rubby No. 6A, Ajibarang
082220399403
mcm.beautycenter

Quick Links
> About Us
> Information
> Treatment

Business Hours
Monday - Thursday
09:00 am - 17:00 pm
Friday
Closed
Saturday
09:00 am - 12:00 pm

fb ig wa

Gambar 3. 14 Wireframe Halaman Diagnosis

Pada halaman diagnosis gambar 3.14, *user* mengisi data diri berupa data nama lengkap, jenis kelamin, dan umur. Selanjutnya pengguna dapat memilih gejala yang masalah kulit wajah yang dialami untuk selanjutnya dilakukan proses penghitungan *similarity* menggunakan algoritma *nearest neighbor*.

3.3.5.7 Wireframe Halaman Hasil Diagnosis

SIMALA

http://127.0.0.1:8000/user/diagnosa/2

[BERANDA](#) [TENTANG](#) [JENIS PERAWATAN](#) [DIAGNOSIS](#)

User

Hasil Diagnosis

Tampilan Hasil Diagnosis
Print

Tanggal & Waktu Diagnosis :

Nama Lengkap :

Jenis Kelamin :

Umur :

MENURUT HASIL DIAGNOSIS SISTEM PAKAR, ANDA MENGALAMI MASALAH KULIT WAJAH Penuaan (Aging) K003, DENGAN NILAI SIMILARITY SEBESAR 0,533

TABEL GEJALA YANG DIPILIH

No.	Kode Gejala	Nama Gejala
1	G002	Kulit cenderung berminyak
2	G005	Kulit cenderung kehilangan elastisitasnya
3	G009	Pori-pori besar
5	G015	Pigmentasi (warna) kulit yang tidak merata

TABEL HASIL PERHITUNGAN METODE CASE BASED REASONING

No.	Kode Kasus	Jenis Masalah Kulit Wajah	Nilai Similarity
1	K001	Jerawat (Acne)	0.087
2	K002	Melasma (Flek)	0
3	K003	Penuaan (Aging)	0.533
4	K004	Melasma (Flek)	0.421
5	K005	Penuaan (Aging)	0.231
6	K006	Jerawat (Acne)	0.631
7	K007	Jerawat (Acne)	0.059
8	K008	Melasma (Flek)	0.067
9	K009	Jerawat (Acne)	0.053
10	K010	Jerawat (Acne)	0.133

Dari data diatas, dapat disimpulkan bahwa pasien tersebut menderita masalah kulit wajah Penuaan Aging K005, dengan nilai similarity ialah 0.533 < nilai threshold, Anda perlu melakukan konsultasi untuk mengetahui jenis treatment apa yang cocok dengan anda.

Apakah anda ingin melakukan konsultasi?

Minta Konsultasi

Our Office

Perum Green Shapphire
Blok Ruby No. 6A, AjiBarang
082220399903
mcm.beauty.center

fb
ig
wa

Quick Link

> About Us
> Information
> Treatment

Business Hour

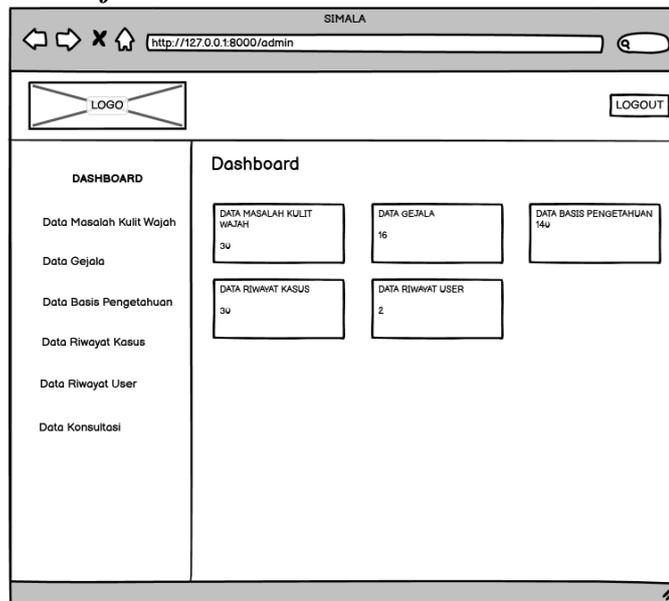
Monday - Thrusda
09:00 am - 17:00 pm
Friday
Closed
Saturday
09:00 am - 12:00 pm

Gambar 3. 15 Wireframe Halaman Hasil Diagnosis

Gambar 3.15 merupakan tampilan hasil diagnosis yang berisi hasil diagnosis dimana terdapat tabel data gejala yang dipilih, tabel hasil perhitungan metode *case based reasoning* disertai *treatment* yang disarankan

dari klinik MCM Beauty Center Ajibarang. Apabila hasil diagnosis kurang dari nilai *threshold* (0,8) saran *treatment* akan dihold oleh pakar (proses *revise*) untuk kemudian pakar akan mengecek *treatment* apa yang sesuai.

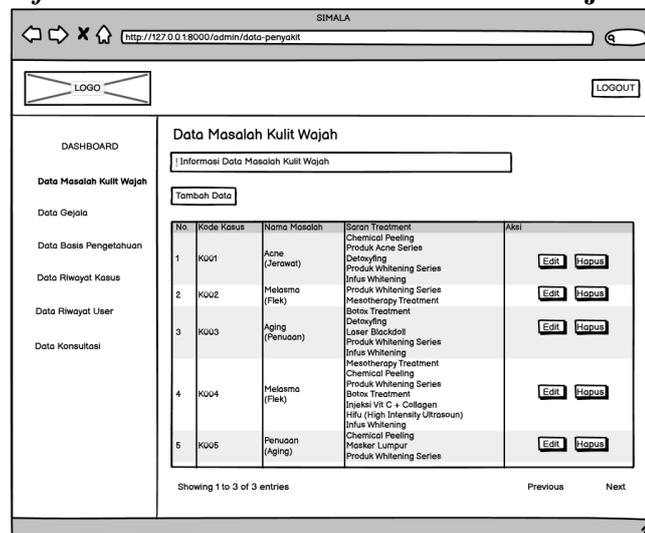
3.3.5.8 Wireframe Halaman Dashboard Admin



Gambar 3.16 Wireframe Halaman Dashboard Admin

Gambar 3.16 adalah tampilan yang muncul setelah admin berhasil *login* ke *dashboard* admin. Di sana, admin dapat melihat data permasalahan kulit wajah, data gejala, data basis pengetahuan, riwayat kasus, riwayat pengguna, dan data konsultasi.

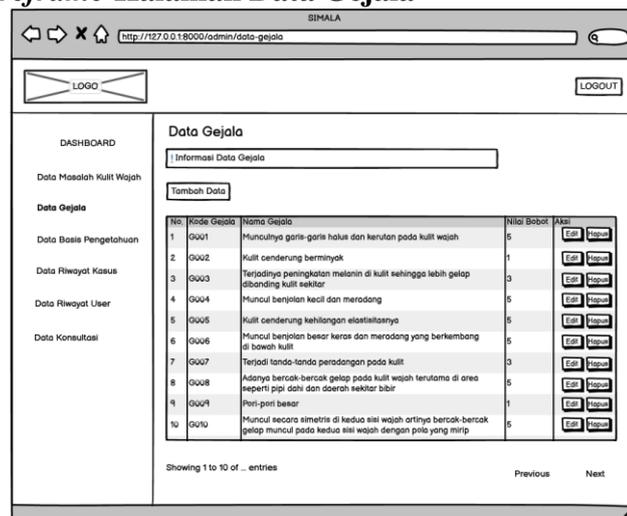
3.3.5.9 Wireframe Halaman Data Masalah Kulit Wajah



Gambar 3. 17 Wireframe Halaman Data Masalah Kulit Wajah

Pada tampilan gambar 3.17 menampilkan kode kasus berupa K001, K002, K003, dst. dengan tiga jenis masalah kulit wajah berupa *Acne* (Jerawat), *Melasma* (Flek), *Aging* (Penuaan) disertai *treatment* yang disarankan, pada tampilan ini admin dapat mengedit data untuk melakukan perubahan serta menambah data jenis masalah kulit wajah beserta saran *treatment*.

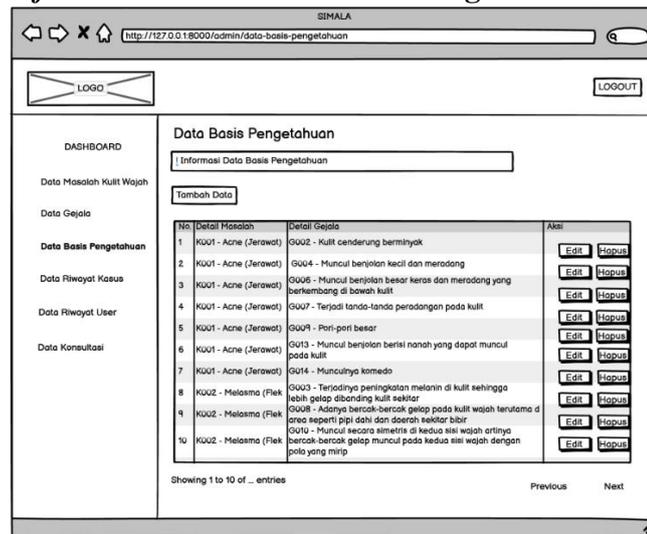
3.3.5.10 Wireframe Halaman Data Gejala



Gambar 3. 18 Wireframe Halaman Data Gejala

Gambar 3.18 merupakan tampilan halaman data gejala berupa kode gejala, nama gejala, dan nilai bobot. Dimana nilai bobot telah ditampilkan pada tabel nilai bobot pakar[5].

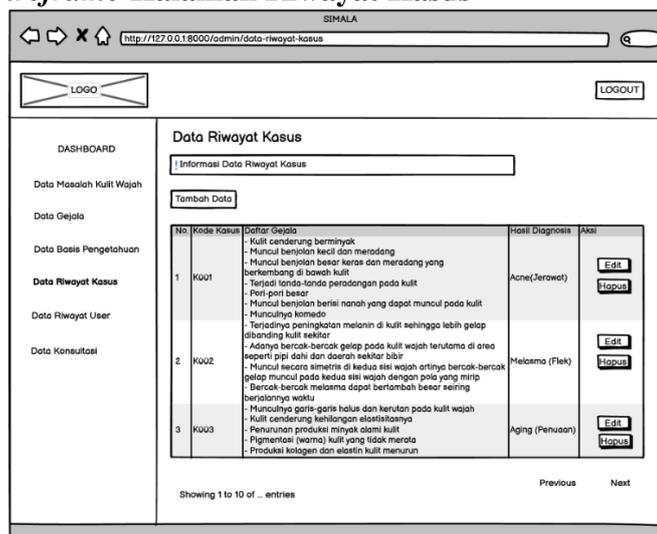
3.3.5.11 Wireframe Halaman Data Basis Pengetahuan



Gambar 3. 19 Wireframe Halaman Data Basis Pengetahuan

Gambar 3.19 menunjukkan tampilan halaman basis pengetahuan, di mana detail masalah kulit wajah dan gejala ditampilkan. Admin dapat menambah dan mengubah data basis pengetahuan berdasarkan data yang diperoleh dari diagnosis pakar.

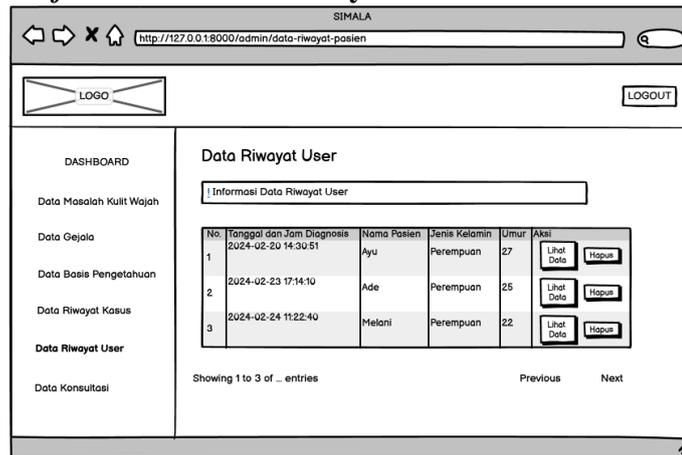
3.3.5.12 Wireframe Halaman Riwayat Kasus



Gambar 3.20 Wireframe Halaman Data Riwayat Kasus

Pada halaman riwayat kasus gambar 3.20 menampilkan beberapa kode kasus di database, daftar gejala dalam bentuk array, serta hasil diagnosis pakar. Admin dapat melakukan *update* data dan hapus data pada halaman riwayat kasus.

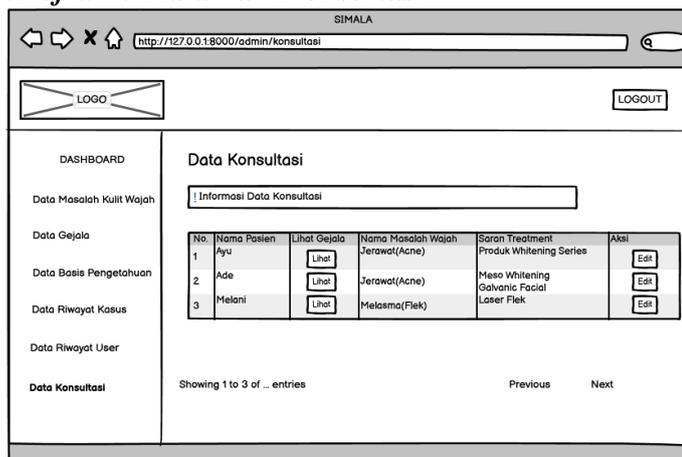
3.3.5.13 Wireframe Halaman Riwayat User



Gambar 3. 21 Wireframe Halaman Data Riwayat User

Pada halaman riwayat *user* gambar 3.21 disini admin dapat melihat detail data pasien berupa tanggal dan jam diagnosis, nama pasien, jenis kelamin, dan umur. Pada bagian ini bisa melihat data riwayat yang telah dilakukan proses diagnosis dari *user*.

3.3.5.14 Wireframe Halaman Konsultasi

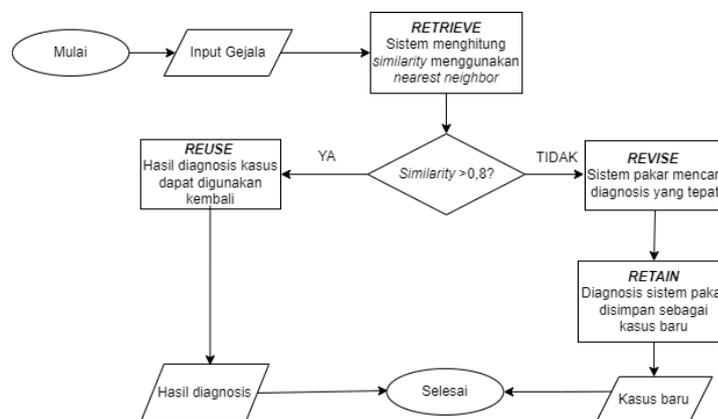


Gambar 3. 22 Wireframe Halaman Data Konsultasi

Pada gambar 3.22 menampilkan data konsultasi, dimana menu ini dilakukan apabila dari user memiliki hasil diagnosis dengan nilai *threshold* < 0,8. Proses *revise* ini pakar akan mengecek untuk melakukan validasi jenis masalah kulit wajah dan saran *treatment* yang dianjurkan di klinik.

3.3.6 Implementasi Sistem

Setelah tahap perancangan sistem selesai, hasilnya akan diterapkan. Bahasa pemrograman PHP *framework* Laravel dan infrastruktur lainnya digunakan penulis pada tahap implementasi sistem secara rinci akan dibahas pada bab 4. Flowchart Alur Sistem di bawah ini akan menjelaskan alur sistem.



Gambar 3.23 Flowchart Alur Sistem[31]

Penjelasan diagram alur sistem pada Gambar 3.23 disajikan di bawah ini. Pertama, diagram alur sistem menunjukkan bahwa gejala yang dialami pengguna harus dipilih dan dimasukkan ke dalam sistem sebagai tindakan pertama yang harus dilakukan. Selanjutnya, sistem akan menjalankan prosedur *Retrieve*, yang melibatkan penggunaan algoritma *nearest neighbor* untuk menentukan tingkat kesamaan gejala. Sistem akan melanjutkan ke proses *Reuse* kasus lama untuk digunakan kembali jika kemiripan gejala yang dihasilkan > 0,8. Hal ini akan memungkinkan pengambilan temuan diagnosis dengan cepat. Sementara itu, sistem akan melanjutkan ke tahap *Revise* dan *Retain* apabila kemiripan gejala yang dihasilkan < 0,8. Proses ini bertujuan memberikan gejala pengguna sebagai kasus baru dengan menambah nilai

pada sumber kasus yang ada pada sistem, kasus-kasus tambahan ini akan meningkatkan kecerdasan sistem.

Berikut adalah *flowchart* dari algoritma *nearest neighbor* yang menggambarkan bagaimana proses mencari kesamaan antara kasus baru saat ini dan kasus sebelumnya. Ini menggunakan prosedur pembobotan.



Gambar 3. 24 *Flowchart* Algoritma *Nearest Neighbor*[32]

Diagram alir pelaksanaan prosedur ditunjukkan pada gambar 3.24 dijelaskan sebagai berikut:

- 1) Pastikan bagaimana setiap atribut kasus diberi bobot
Untuk mencapai hal ini, para ahli dikonsultasikan untuk menentukan bagaimana mempertimbangkan fitur-fitur kasus tersebut.
- 2) Tentukan seberapa mirip kasus baru dengan kasus sebelumnya.
Gunakanlah perhitungan kemiripan (*similarity*) untuk menentukan seberapa mirip kasus baru dan kasus lama.
- 3) Mengurutkan kasus lama
Setiap kasus sebelumnya yang telah dihitung *similarity* diurutkan kasusnya mulai dari yang tingkat kemiripan paling tinggi.

4) Pemilihan solusi untuk kasus baru

Solusi kasus baru berasal dari solusi kasus lama dengan nilai kemiripan paling tinggi

3.3.7 Pengujian Sistem

Pengujian sistem yang baru dikembangkan dilakukan berikutnya, setelah implementasi sistem berhasil. Pengujian *black box*, yang menguji aplikasi hanya dari antarmukanya dan menilai kemampuan fungsionalnya, adalah teknik yang digunakan dalam pengujian sistem. Sistem akan diuji untuk memastikan bahwa fungsionalitasnya berjalan dengan baik. Setelah itu, dilakukan pengujian presisi untuk memverifikasi kesesuaian data yang diperoleh dari pakar dengan hasil yang dihasilkan oleh sistem pakar di *website*. Rumus berikut digunakan saat melakukan pengujian.

$$\text{Presisi} = \frac{N_{\text{benar}}}{N_{\text{total}}} \times 100\% \quad (3.1)$$

3.3.8 Kesimpulan

Pada tahap ini, kesimpulan akan dibuat untuk mengambil inti dari penelitian dan memberikan pemahaman yang lengkap. Jika tujuan dan hasilnya sesuai, penelitian ini dapat digunakan sebagai dasar untuk penelitian selanjutnya.