BAB 3
ANALISIS DAN PERANCANGAN SISTEM

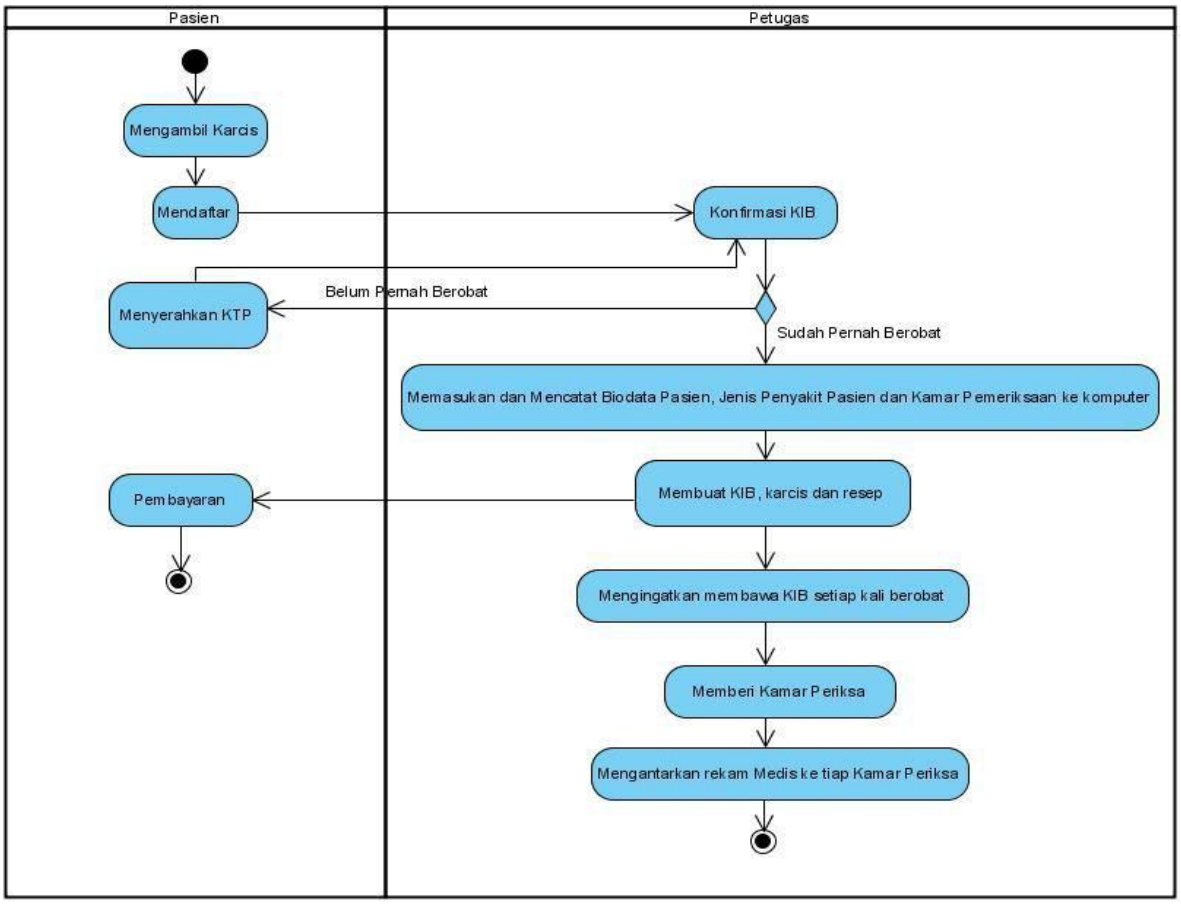
3.1 Analisis Sistem

Analisis dilakukan dengan tujuan untuk mengidentifikasi dan mengevaluasi seluruh komponen yang terkait dengan sistem yang akan dibangun. Tahap analisis merupakan tahap yang kritis dan sangat penting, karena kesalahan pada tahap ini akan menyebabkan kesalahan pada tahap selanjutnya. Tahap analisis sistem dilakukan dengan cara menguraikan suatu sistem yang utuh kedalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan-permasalahan sehingga ditemukan kelemahan dan keuntungan pada sistem tersebut, sehingga dalam membangun aplikasi menjadi lebih mudah dari hasil analisis sistem yang lama. Analisis sistem ini akan menghasilkan beberapa data dan fakta yang kemudian akan dijadikan bahan dalam pembuatan aplikasi dokter *virtual*

3.1.1 Analisis Sistem Yang Sedang Berjalan (Prosedur Berobat Jalan Pasien)

Proses bisnis yang sedang berjalan untuk pasien beribat jalan pada klinik Iin Hatinah, saat ini dimulai ketika pasien datang ke klinik untuk berubat atau berkonsultasi ke pada dokter. jika pasien baru maka akan melakukan pendaftaran (*registrasi*) dengan megisi data diri dibagian admistrasi untuk mendapatkan kartu pasien dan kartu berobat.setelah melakukan pendaftaran (*registrasi*) petugas admistrasi akan mencatat data pasien ke dalam kartu pasien dan pada kartu berobat. jika pasien merupakan pasien lama, maka pasien datang dengan membawa kartu pasien untuk melakukan pendaftaran di bagian admistrasi, maka petugas admistrasi akan mengambilkan kartu berobat atas nama pasien tersebut dan di tempatkan ke dalam antrian yang sudah ada. kemudian pasien akan menunggu diruangan tunggu sampai nama pasien di panggil.pada saat pasien di panggil maka pasien masuk dengan membawa kartu berobat ke dalam ruangan dokter dan menyerahkan kartu tersebut kepada dokter.

Setelah dokter mendapatkan kartu berobat dari pasien, maka dokter akan melakukan pemeriksaan terhadap pasien atas keluhan yang disampaikan oleh pasien tersebut, setelah pasien diperiksa dan berkonsultasi dengan dokter, maka dokter mencatat diagnosa dan resep kedalam kartu berobat tersebut. pasien keluar dangen membawa kartu berobat dan memberikannnya kepada bagian apotek untuk menebus obat, sementara pasien menunggu obat disiapkan oleh petugas apotek, setelah obat siap, maka petugas apotek memanggil nama pasien dan memberikan obat yang sesuai dengan resep yang diberikanm pesien akan melakukan pembayaran dan petugas apotek akan memberikan bukti pembayaran.



Gambar 3.1. Activity Diagram Sistem yang Sedang Berjalan

3.1.2 Analisis Masalah

Informasi mengenai kesehatan adalah bagian sangat penting dalam kehidupan sehari-hari, dimana informasi kesehatan tersebut diperoleh salah satunya dengan kosultasi langsung dengan dokter. Berdasarkan wawancara yang dilakukan terhadap beberapa pasien di klinik dr Iin Hartinh pasien memperoleh informasi kesehatan dengan cara berkunjung ke klinik dan berkonsultasi langsung terhadap dokter. Penyampaian informasi kesehatan tersebut dirasa kurang karena keterbatasan oleh waktu jam buka klinik, sedangkan banyak pasien sangat membutuhkan informasi kesehatan dan konsultasi kesehatan dengan cepet dan tidak mau terikat oleh waktu buka klinik, bahkan pasien mengalami masalah Kesehatan disaat klinik sudah tutup, dan membutuhkan konsultasi dokter. Dalam penyampaian informasi kesehatan yang tidak terikat oleh waktu buka klinik, klinik dr Iin Hartinah memerlukan suatu alat media layanan informasi kesehatan yang dapat merespon setiap pertanyaan pasien tanpa ada keterbatasan waktu dan jumlah dokter.

3.1.3 Solusi yang Dibangun

Berdasarkan permasalahan yang dihadapi di Klinik dr Iin Hartinah, maka solusi yang akan dibangun adalah membuat aplikasi *chatbot* dengan nama dokter *virtual*. Aplikasi ini dapat berkomunikasi langsung dengan pengguna menggunakan bahasa sehari-hari melalui pertanyaan yang diberikan oleh pengguna. Aplikasi ini dibangun menggunakan pendekatan *Natural Language Processing* dengan *fuzzy string matching* sebagai metode untuk proses penalaran kalimat/pertanyaan.

Chatbot yang dibangun memiliki kemampuan memberikan kemudahan kepada pengunjung untuk mendapatkan layanan layaknya berkonsultasi dengan seorang dokter.

Adapun *chatbot* dokter *virtual* yanng dibangun memiliki karakteristik sebagai berikut:

1. Program mampu membaca koleksi pertanyaan dan jawaban, dan menampilkan dialog percakapan pengguna dan *chatbot* dokter *virtual*.
2. Semua pertanyaan dan jawaban dalam Bahasa Indonesia

3. Pencocokan yang dilakukan adalah dalam satu kesatuan *string* pertanyaan utuh (misal "Mengapa saya sering sakit kepala?"), bukan kata per kata ("Mengapa", "saya", "sering", "sakit", "kepala").

3.1.4 Deskripsi Sistem

Sistem yang dibuat dalam skripsi ini adalah sistem yang dapat mengatasi masalah yang telah disebutkan diatas. Sistem yang diberi nama dokter virtual ini adalah suatu sistem yang selain dapat berinteraksi menggunakan bahasa manusia dan melakukan percakapan.

Dokter *virtual* dibuat dengan menggunakan penerapan sistem *Chatting*, dimana pengguna dapat berinteraksi langsung dengan *Chatbot*. Pengguna dapat berinteraksi dengan sistem dan dapat mengajukan pertanyaan menggunakan bahasa Indonesia. Bila pola kalimat dikenali maka sistem akan mencari apakah pola kalimat yang dimaksud oleh pengguna cocok dengan pola kalimat yang ada pada basis data. Pola kalimat yang tidak sesuai akan mengakibatkan dokter *virtual* tidak dapat mengolah pertanyaan dan memberikan jawaban yang tidak diharapkan.

Tabel berikut menunjukkan contoh penghitungan tingkat kecocokan antar pertanyaan yang dilakukan sistem.

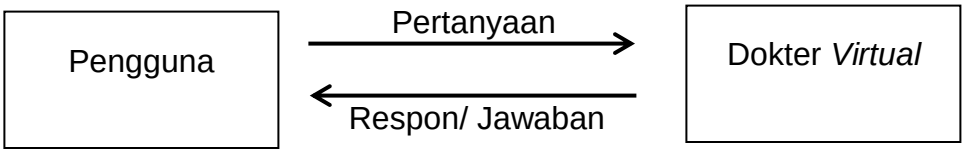
Tabel 3.2. Tabel perhitungan kecocokan pertanyaan

Pertanyaan dari pengguna	Pertanyaan yang tersimpan	Tingkat Kemiripan
Apakah chatbot itu ?	Apakah chatbot itu ?	100%
Apakah chatbot ?	Apakah chatbot itu manusia ?	14/22 = 63,6%
Apakah chatbot itu ?	Apakah chatbot ?	100%

Dokter virtual memberikan jawaban berdasarkan basis pengetahuan yang dimilikinya. Semakin banyak basis pengetahuan yang dimiliki, maka dokter *virtual* akan semakin pintar.

Basis pengetahuan dokter virtual menggunakan dokumen basis data MySQL. Basis pengetahuan tersebut bisa ditambahkan oleh seorang administrator yang dalam hal ini disebut sebagai *botmaster*.

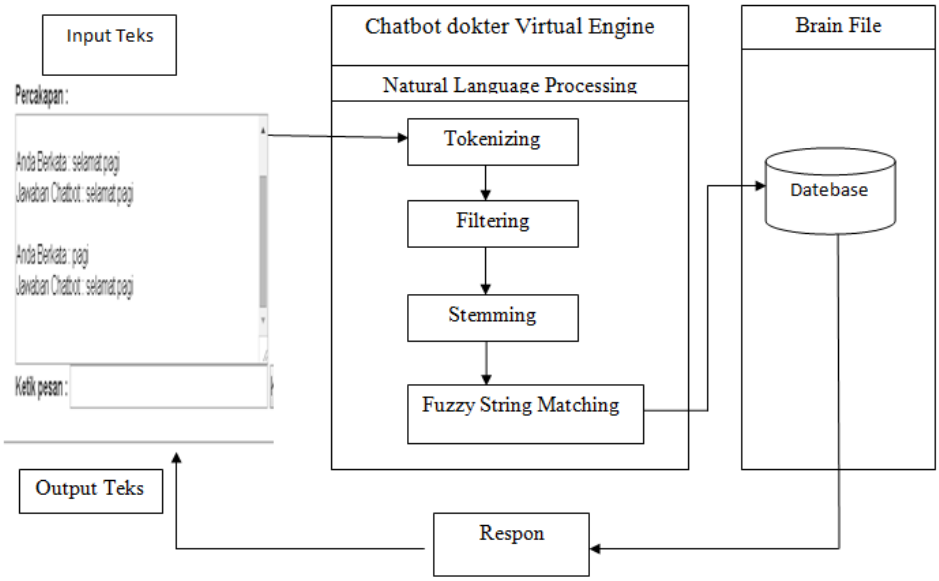
Secara umum konsep dasar komunikasi antara pengguna dengan dokter *virtual* bisa digambarkan sebagai berikut:



Gambar 3.2. Alur komunikasi pengguna dengan Dokter *Virtual*

3.1.5 Arsitektur Sistem

Sistem terdiri atas 3 bagian pokok, yaitu bagian pemahaman kalimat *input*, bagian untuk memberikan jawaban dan bagian untuk pengelolaan basis pengetahuan. Pada saat dokter *virtual* pertama kali dijalankan, maka akan muncul layar front page yang menampilkan profil dari Klinik dr Iin Hartinah. Pada halaman profil tersebut terdapat *menu* untuk membuka aplikasi *chatbot* dokter *virtual*. Setelah itu dokter *virtual* akan memulai percakapan dengan menyapa dan menawarkan bantuan kepada *user*. Kemudian *user* akan berbicara kepada dokter *virtual* dengan memberikan *input* berupa suatu kalimat. Kemudian *input* yang masih dalam bentuk kalimat akan diolah dan dicari pola yang cocok dengan basis pengetahuan yang dimiliki oleh aplikasi dokter virtual. Jika pola sudah ditemukan, maka proses selanjutnya adalah mencari jawaban dari *input* tersebut. Setelah jawaban ditemukan maka dokter *virtual* menampilkan jawaban dalam layar tampilan sebagai *output*.



Gambar 3.2 Arsitektur *Chatbot* Dokter *Virtual*

3.1.6 Analisis Data

Berdasarkan batasan masalah yang telah diuraikan terdahulu, aplikasi dokter *virtual* yang akan dibangun mampu untuk memberikan informasi mengenai Klinik dr Iin Hartinah dan memberikan layanan konsultasi masalah kesehatan kepada pengguna. Adapun detail informasi yang dapat diberikan oleh *chatbot* dokter *virtual* adalah sebagai berikut:

1. Informasi mengenai Klinik dr Iin Hartinah
    a. Klinik dr Iin Hartinah
    b. Alamat Klinik
    c. Waktu pelayanan pasien
    d. Fasilitas layanan
    e. Dokter yang bertugas
    f. Informasi dan prosedur registrasi pasien
2. Layanan konsultasi masalah kesehatan

Data layanan konsultasi mengenai masalah kesehatan dan informasi penyakit disesuaikan dengan yang terdapat pada buku Kapita Selekta Kedokteran Jilid Pertama.

3.1.7 Analisis Metode

3.1.7.1 Tokenizing (Parsing)

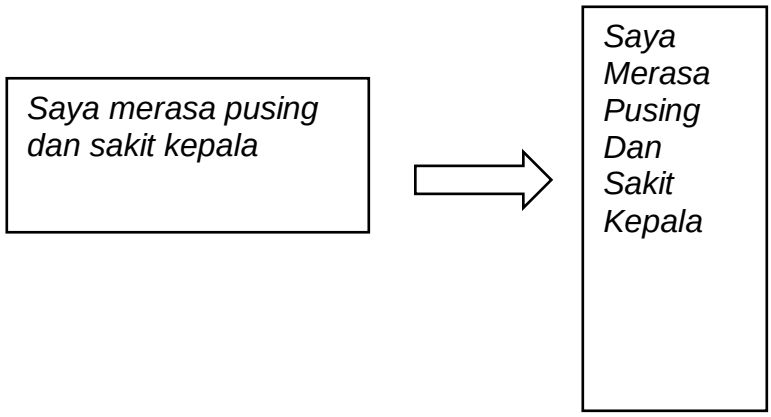
Proses ini akan melakukan pengecekan dari karakter pertama sampai dengan karakter terakhir. Apabila karakter ke-i bukan merupakan pemenggal kata maka akan ditambahkan dengan karakter selanjutnya. Karakter pemenggal kata ini contohnya seperti tanda baca atau spasi.

Berikut merupakan langkah-langkah umum pilah kata dalam contoh kalimat (pertanyaan) dari deskripsi yang berisi kata-kata " saya merasa pusing dan sakit kepala,?"

Saya merasa pusing dan sakit kepala ?

1. Memotong setiap kata dalam teks, dan mengubahnya menjadi huruf kecil.
2. Hanya huruf ‘a’ sampai ‘z’ yang diterima, sedangkan karakter selain huruf dihilangkan.
3. Hasilnya adalah kata-kata yang merupakan penyusun kalimat yang dimasukkan.

Gambaran tahap tokenizing dapat dilihat pada gambar 3.3



(hasil *tokenizing*)

Gambar 3.3 Tahap *Tokenizing*

Proses parsing merupakan proses untuk menterjemahkan masukan dari pengguna agar dapat dimengerti oleh sistem. Parsing merujuk pada proses pengenalan *token* yang terdapat dalam rangkaian teks.

Kalimat masukan pengguna yang dimasukan akan dianggap sebagai kata-kata yang harus ada pada data yang akan dicari. Aturan proses penguraian kalimat yang dimiliki sistem dapat menggunakan konfigurasi *Finite State Automata*. Konfigurasi FSA dapat dinyatakan dengan Q = Himpunan State, ∑ = Himpunan Simbol Input, S = State Awal, F = *State* Akhir. Konfigurasi FSA pada sistem ini dapat dinyatakan sebagai berikut :

Q = { Q0, Q1}

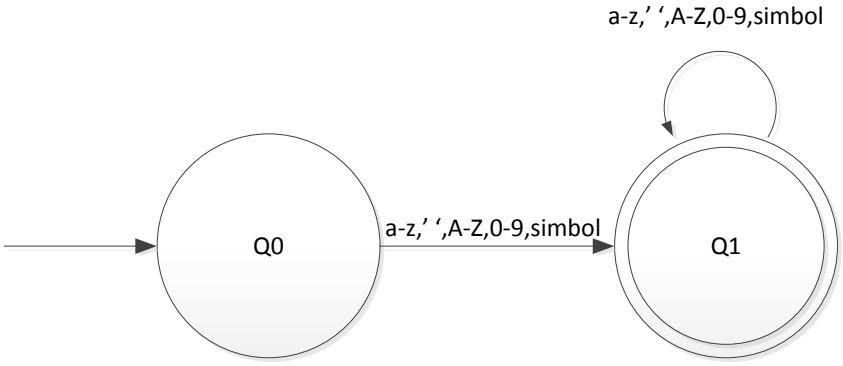
∑ = { a-z,' ',A-Z,0-9,symbol },

dimana symbol = [!|@|#|$|%|^|&|*|(|)|_|+|}|{|?|>|<|.|,|;|:]

S = { Q0 }

F = { Q1 }

Aturan proses penguraian kalimat yang dimiliki sistem dapat dilihat pada gambar 3.4



Gambar 3. 4 *Finite State Automata* Proses *Parsing*

Fungsi *transisi* yang didapat adalah sebagai berikut :

δ(Q0, a-z|' '|A-Z|0-9|simbol) = Q1

δ(Q1, a-z|' '|A-Z|0-9|simbol) = Q1

Fungsi tersebut dapat dilihat pada tabel 3.3

Tabel 3.3 Tabel Transisi

δ	a-z|' '|A-Z|0-9|simbol
Q0	Q1
Q1	Q1

Dari diagram state gambar 3.11 , didapat aturan produksi sebagai berikut:

S ->aA..zA| AA..ZA | 0A-9A|simbol|<space>A|a..Z|A..Z|<space>

A ->aA..zA| AA..ZA |0A-9A|simbol|<space>A|a..Z|A..Z|<space>

Dimana secara formal dapat dinyatakan dengan V = himpunan simbol variabel/non terminal, T = himpunan simbol terminal, P = kumpulan aturan produksi, S = simbol awal.

V = {S, A}

T = {a-z,' ' A-Z,0-9,simbol}

P = {S->aA..zA|AA..ZA|0A..9A|simbol|<space>A|a..Z|A..Z|<space>,

A->aA..zA| AA..ZA |0A..9A|simbol|<space>A|a..Z|A..Z|<space>}

S = S

Aturan produksi diatas menggunakan aturan *left to right* sehingga jika pengunjung memasukan pertanyaan " Apa itu demam ?!", maka proses akan dilakukan dari depan (*left*) ke belakang (*right*), seperti berikut :

S ->aA

S ->apA

S ->apaA

S ->apa iA

S ->apa itA

S ->apa ituA

S ->apa itu dA

S ->apa itu deA

S ->apa itu demA

S ->apa itu demaA

S ->apa itu demamA

S ->apa itu demam

Pada proses di atas setiap kalimat atau kata akan dipecah menjadi perbagian karakter dalam bentuk *array.*

3.1.7.2 Filtering

Filtering proses yaitu mengambil kata-kata penting dari hasil *token*. Bisa menggunakan algoritma *stoplist* (membuang kata yang kurang penting) atau *word list* (menyimpan kata penting). Sistem ini menggunakan metode *stoplist* yaitu penghilangan kata tidak penting *(stopword)* pada deskripsi melalui pengecekan kata-kata hasil *token* deskripsi apakah termasuk di dalam daftar kata tidak penting *(stop list)* atau tidak. Jika termasuk di dalam *stoplist* maka kata-kata tersebut akan di-*remove* dari deskripsi sehingga kata-kata yang tersisa di dalam deskripsi di anggap sebagai kata-kata penting atau *keywords (pattern)*. Tahapan proses *filtering* adalah sebagai berikut :

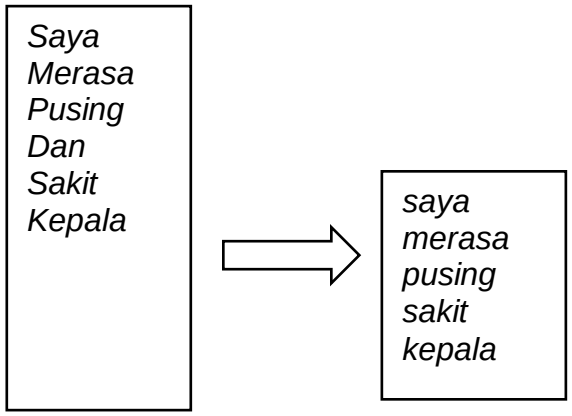
1. Kata hasil token *stemming* dibandingkan dengan tabel *stopword*.
2. Dilakukan pengecekan apakah token sama dengan tabel *stopword* atau tidak.

3. Jika token sama dengan tabel stopword maka akan di-*remove.*
4. Jika token tidak sama dengan tabel stopword akan ditampilkan. Yaitu menghasilkan token hasil *filtering* yang termasuk kata penting(*keyword*)

Contoh dari tahap *filtering* adalah sebagai berikut:

Stopwords : saya, dan, (kata yang akan diremove)

Tahapan dalam *filtering* dapat dilihat pada gambar 3.5



(hasil *tokenizing*) (*keyword* hasil *filter*)

Gambar 3.5 Tahap *Filtering*

Proses *filtering* merupakan proses untuk melakukan pengecekan kata-kata hasil token apakah termasuk didalam daftar kata tidak penting (*stoplist*). Jika termasuk dalam *stoplist* maka kata-kata tersebut akan dihilangkan sehingga kata-kata yang tersisa dianggap sebagai kata-kata penting atau *keyword (pattern)*. Konfigurasi FSA pada *filtering* dapat dinyatakan sebagai berikut :

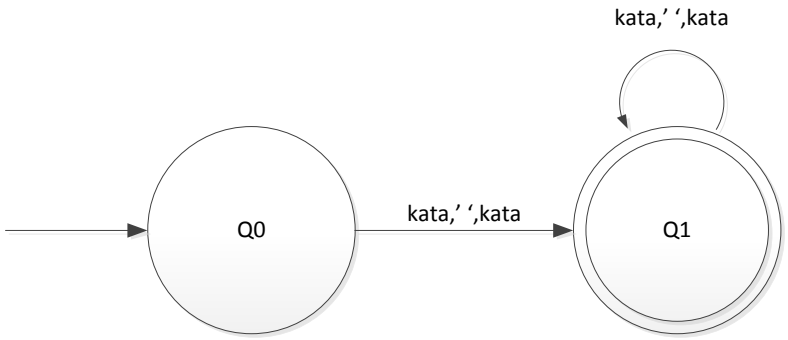
Q = { Q0, Q1}

∑ = { kata,' ',kata },

S = { Q0 }

F = { Q1 }

Aturan proses penguraian kalimat yang dimiliki sistem dapat dilihat pada gambar 3.6



Gambar 3. 6 *Finite State Automata* Proses *Filtering*

Fungsi *transisi* yang didapat adalah sebagai berikut :

δ(Q0, kata|' '|kata) = Q1

δ(Q1, kata|' '|kata) = Q1

Fungsi tersebut dapat dilihat pada tabel 3.4

Tabel 3.4 Tabel Transisi

δ	kata|' '|kata
Q0	Q1
Q1	Q1

Dari diagram state gambar 3.6 , didapat aturan produksi sebagai berikut:

S ->kata|<space>kata|<space>

A ->kata stoplist

Dimana secara formal dapat dinyatakan dengan V = himpunan simbol variabel/non terminal, T = himpunan simbol terminal, P = kumpulan aturan produksi, S = simbol awal.

V = {S, A}

T = {kata,’ ‘,kata}

P = {kata|<space>|kata|,

A->kata stoplist}

S = S

Dari proses penghilangan daftar kata tidak penting diatas, maka tersisa kata “ pusing sakit kepala” yang dianggap sebagai kata-kata penting atau *keywords*. *Keyword* yang telah didapat dicocokan dengan *brainfile* yang ada pada database. *Keyword* tersebut nantinya yang akan mengacu menjadi jawaban terhadap pertanyaan pengunjung. Daftar kata stoplist dan *brainfile chatbot* dapat dilihat pada tabel 3.5

Tabel 3.5 Daftar Kata *Stopword*

No	kata stoplist	No	kata stoplist	no	kata stoplist	no	kata stoplist
1	Yang	26	Pernah	51	Saya	76	Mengapa
2	Mampu	27	Setiap	52	Off	77	Akan
3	Tentang	28	Untuk	53	Sering	78	Dengan
4	Di	29	Dari	54	Pada	79	Letak
5	Setelah	30	Mendapatkan	55	Hanya	80	Belum
6	Semua	31	Punya	56	Atau	81	Anda
7	Hamper	32	Telah	57	Kita	82	Alamat
8	Juga	33	Memiliki	58	Sendiri	83	Jalan
9	Antara	34	Dia	59	Agak	84	Lokasi
10	Dan	35	Miliknya	60	Kata	85	Sebuah
11	Ada	36	Bagaimana	61	Begitu	86	Atas
12	Seperti	37	bagaimanapun	62	Beberapa	87	Menurut
13	Jadi	38	I	63	Mereka	88	Sesuai
14	Karena	39	Jika	64	Kemudian	89	Seberang
15	Sudah	40	Ke	65	Sana	90	sebenarnya
16	Tetapi	41	Dalam	66	Ini	91	Sekali
17	Oleh	42	Itu	67	Sungguh	92	Lagi
18	Bisa	43	Sama	68	Yang	92	Donk
19	Tidak	44	Paling	69	Adalah	94	Terhadap
20	Saying	45	Biarkan	70	Kami	95	Mungkin
21	Melakukannya	46	Mungkin	71	Apa	96	Meskipun
22	Lakukan	47	Aku	72	Kapan	97	Selalu
23	Memang	48	Sebagian	73	Mana	98	Apapun
24	Baik	49	Besar	74	Berada	99	Siapapun
25	Lain	50	Harus	75	Siapa	100	Saja

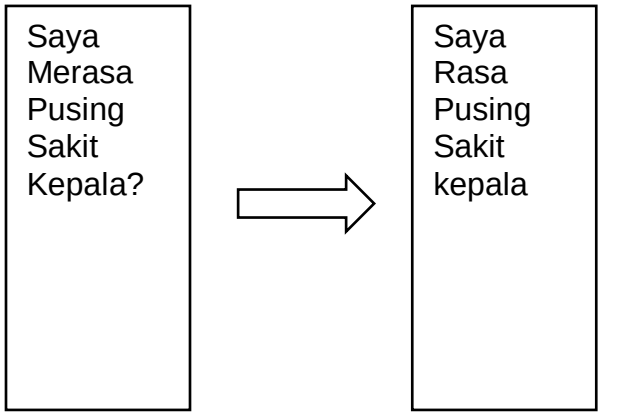
Tahapan pencocokan token dengan *stopword* pada *filtering* adalah sebagai berikut :

1. Kata hasil token *stemming* dibandingkan dengan tabel *stopword.*
2. Dilakukan pengecekan apakah token sama dengan tabel *stopword* atau tidak.
3. Jika token sama dengan tabel stoplist maka akan di-*remove.*
4. Jika token tidak sama dengan tabel *stoplist* akan ditampilkan. Yaitu menghasilkan token hasil *filtering* yang termasuk kata penting(*keyword*)

3.1.7.3 Stemming

Merupakan pengelolaan *keyword* menjadi *keyword* yang utuh yaitu dengan menghilangkan imbuhan seperti diantaranya "yang", "di", "ke", "me", "meng", "kan". Penguraian dari suatu kata menjadi bentuk kata dasarnya *(stem)*. Untuk lebih jelasnya tahapan proses *stemming* adalah sebagai berikut :

1. Token hasil *tokenizing* diperiksa apakah mengandung imbuhan atau tidak.
2. Jika terdapat imbuhan maka akan dilakukan pembuangan imbuhan, terus berulang sampai tidak mengandung imbuhan.
3. Jika tidak mengandung imbuhan maka akan ditampilkan.



Gambar 3.7 Tahap *Stemming*

3.1.7.4 Metode fuzzy string matching

Metode fuzzy string matching biasa juga disebut metode *inexact string matching* adalah metode pencocokan yang digunakan untuk mencari *string* yang memiliki kesamaan dengan string tertampung di kamus, meskipun tidak sama persisi dalam susunan karakternya. Untuk mendapatkan kesamaan ini dibutuhkan suatu fungsi yang sering disebut dengan *similarity function*, *similarity functio*n adalah fungsi yang akan bertugas memutuskan string hasil pencarian jika ditemukan string hasil pendekatan *(aproksimasi).* Similarity function pada data *string* didahului dengan normalisasi string sesuai dengan aturan *translasi Q-gram* maupun eliminasi duplikasi

karakter setelah dilakukannya tranlasi *Q-gram. Q-gram* adalah sususan beberapa huruf yang berurutan. Sedang *eliminasi* duplikasi karakter adalah menghilangkan karakter yang memiliki pengulangan pada suatu *string*.

Algoritma (langkah-langkah) *fuzzy string matching* dalam *implementasi web chatbot* situs klinik, adalah sebagai berikut.

1. Menormalisasikan karakter suatu data *string* yang akan dibaca dan dipahami oleh chatbot.
   - Bila karakter yang dibaca pada *string* yang dibandingkan memiliki Q-gram (urutan karakter) yang ada pada suatu tabel aturan Q-gram maka Q-gram pada string tersebut akan ditranslasikan ke dalam bentuk normal karakter tersebut, proses *translasi* Q-gram ini akan dilakukan pada suatu data *string* yang dibaca selama ditemui adanya suatu urutan karakter yang terdapat pada tabel aturan Q-gram, bila kemudian terdapat karakter berdekatan yang berulan maka salah satu karakter tersebut akan dieliminasi.

1	2	3	4	5	6	7	8
V	i	I	T	A	M	I	N

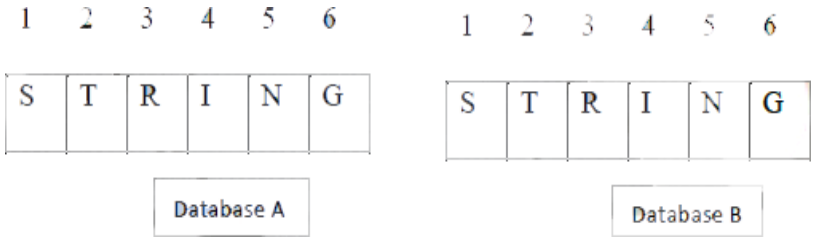
1	2	3	4	5	6	7
V	I	T	A	M	I	N

   - Bila karakter yang dibaca pada *string* yang dibandingkan memiliki Q-gram (urutan karakter) yang tidak terdapat pada suatu tabel aturan Q-gram maka dapat disimpulkan string tersebut telah memiliki bentuk normal urutan karakter yang tidak perlu ditranslasikan.

1	2	3	4	5
T	U	B	U	H

- Dikarenakan Tubuh merupakan urutan karakter yang tidak terdapat pada tabel Q-gram maka urutan-urutan karakter yang terdapat pada *string* tersebut tidak perlu ditranslasikan.

2. Penghitungan jumlah karakter dari data *string* yang dicocokan pada masing-masing kata.



3. Bila bentuk suatu *string* yang dibaca telah memiliki bentuk normal dari *string* tersebut maka proses membandingkan kecocokan data string dapat dilakukan dengan memeriksa kecocokan tiap-tiap karakter pada data *string* yang dibandingkan.

   - Bila karakter yang dibandingkan memiliki kecocokan atau persamaan karakter maka proses pemeriksaan kecocokan karakter akan berlanjut hingga pada karakter terakhir jumlah *string* tersebut.
   - Bila kemudian ditemui ketidakcocokan karakter pada *string* yang dibandingkan maka dapat disimpulkan kedua string yang dibandingkan tidak memiliki kecocokan sehingga proses pencocokan pun tidak berlanjut (dinyatakan *string* yang dibandingkan tidak cocok).

4. pencocokan kesamaan teks dengan membaca karakter sebelah kiri string yang berindeks-I pada masing-masing kata.
   - Bila karakter pada data masing-masing kamus memiliki kesamaan, maka pencocokan string akan berpindah ke arah kanan karakter *string* selanjutnya, hingga akhir dari *string* tersebut.
   - Sedangkan bila karakter yang dicocokan tidak memiliki kesamaan antar
   - karakter pada masing-masing kata, maka *string* tersebut tidak kembali dicocokan (dinyatakan tidak cocok).
5. Karena adanya karakter yang sama hingga pembacaan pada karakter terakhir antara kedua string pada masing-masing kata, maka dapat disimpulkan data yang dibandingkan memiliki kecocokan.
6. Karena adanya karakter yang sama hingga pembacaan pada karakter terakhir antara kedua *string* pada masing-masing kata, maka dapat disimpulkan data yang dibandingkan memiliki kecocokan.

Multivitamin	Vitamin

7. Untuk menghitung persentase kecocokan antara kedua string yang dibandingkan, langkah pertama adalah menentukan *string* "pola". *String* "pola" diambil dari jumlah karakter terbanyak dari string yang dibandingkan, misalnya :
8. Menentukan nilai minimum untuk nilai pinalti. Nilai penalti adalah nilai yang digunakan jika apabila karakter ditemukan pada *string* "pola", yaitu :

   Nilai Penalti Minimum = $\frac{1}{Jumlah\ Karakter\ String\ "Pola"}$

   Nilai Penalti Minimum = $\frac{1}{12}$

9. Menentukan lokasi karakter *string* yang dibandingkan dengan lokasi karakter pada string "pola", misalnya :'

MULTIVITAMIN, karakter “M” , “I”, “T”,”V”,”A ” dan ”N” yang sama dengan karakter pada string “pola”, dan khusus untuk karakter “I” ditemukan sebanyak 2 posisi di string “pola” VITAMIN

10. Karakter-karakter yang ditemukan tersebut kemudian dihitung penaltinya, langkah pertama yaitu dengan menghitung selisih jarak karakter string dengan karakter pada string “pola”, yaitu :

    Jarak = abs (lokasi karakter string – lokasi karakter *string* “pola”)

    Untuk karakter “M”(1), jarak = 1-5 = 4;

    Untuk karakter “M” (2), jarak = 10-5 = 5;

    Untuk karater “T” (1), jarak = 4 – 3 = 1;

    Untuk karater “T” (2), jarak = 8 – 3 = 5;

    Untuk karater “I” (1), jarak = 5 – 2 = 3;

    Untuk karater “I” (2), jarak = 7 – 2 = 5;

    Untuk karater “I” (3), jarak = 11– 2 = 9;

    Untuk karater “I” (4), jarak = 5 – 6 = 1;

    Untuk karater “I” (5), jarak = 7 – 6 = 1;

    Untuk karater “I” (6), jarak = 11 – 6 = 5;

    Untuk karater “V”, jarak = 6 – 1 = 5;

    Untuk karater “A”, jarak = 9 – 4 = 5;

    Untuk karater “N”, jarak = 12 – 7 = 5

11. Menghitung nilai penalti dari setiap karakter yang ditemukan pada string “pola”, yaitu :

    Nilai Penalti = jarak x nilai penalti minimum

    Untuk karakter “M”(1), Nilai Penalti = 4 x (1/12) = 0.33;

    Untuk karakter “M” (2), Nilai Penalti = 5 x (1/12) = 0.42;

    Untuk karakter “T”(1), Nilai Penalti = 1 x (1/12) = 0.08;

    Untuk karakter “T”(2), Nilai Penalti = 5 x (1/12) = 0.42;

    Untuk karakter “I”(1), Nilai Penalti = 3 x (1/12) = 0.25;

Untuk karakter “I”(2), Nilai Penalti = 5 x (1/12) =0.42;

Untuk karakter “I”(3), Nilai Penalti = 9 x (1/12) = 0.75;

Untuk karakter “I”(4), Nilai Penalti = 1 x (1/12) =0.08;

Untuk karakter “I”(5), Nilai Penalti = 1 x (1/12) =0.08;

Untuk karakter “I”(6), Nilai Penalti = 5 x (1/12) = 0.42;

Untuk karakter “V”, Nilai Penalti = 5 x (1/12) = 0.42;

Untuk karakter “A”, Nilai Penalti = 5 x (1/12) =0.42;

Untuk karakter “N”, Nilai Penalti = 5 x (1/12) =0.42;

12. Menghitung nilai persentase dari kecocokan kedua buah *string* yang dibandingkan, yaitu :

$$\textbf{Nilai Persentase} = \frac{\sum_{n=1}^{x} \mathbf{1} - \textit{nilai penalti}}{\textit{jumlah karakter string “pola”}} \textbf{ x 100\%}$$

Keterangan : x = jumlah karakter *string* yang dibandingkan.

Hasilnya adalah sebagai berikut :

$$\textbf{Nilai Persentase} = \frac{\begin{array}{c}(\mathbf{1}-0.33)+(\mathbf{1}-\mathbf{0{,}42})+(\mathbf{1}-\mathbf{0.08})+(\mathbf{1}-\mathbf{0{,}42}) \\ +(\mathbf{1}-\mathbf{0{,}25})+(\mathbf{1}-\mathbf{0{,}42})+(\mathbf{1}-\mathbf{0{,}75})+(\mathbf{1}-\mathbf{0{,}08}) \\ +(\mathbf{1}-\mathbf{0{,}08})+(\mathbf{1}-\mathbf{0{,}42})+(\mathbf{1}-\mathbf{0{,}42})+(\mathbf{1}-\mathbf{0{,}42})\end{array}}{\mathbf{12}} \textbf{x 100 = 65,91\% kecocokan}$$

Dari hasil perhitungan di atas diketahui nilai kecocokan antara 2 buah *string* yang dibandingkan yaitu MULTIVITAMIN dan VITAMIN adalah **66.91%** kecocokan.

3.2 Analisis Kebutuhan Non Fungsional

Kebutuhan non fungsional adalah usulan yang direkomendasikan kepada pengguna agar perangkat lunak yang akan dibangun menjadi *user friendly* dan perangkat keras yang mendukung secara maksimal terhadap kinerja perangkat lunak.

3.2.1 Analisis Kebutuhan Perangkat Lunak

Spesifikasi perangkat lunak yang dibutuhkan agar sistem dapat berjalan dengan baik sebagia berikut:

1. Admin dan Pengembang
   a. WAMP Server sebagai *Web Server*
   b. Adobe Dreamweaver sebagai editor PHP
   c. MySQL sebagai DBMS
   d. Tool antarmuka design Microsoft Visio 2007
   e. Google Chrome dan Mozila Firefox sebagai browser

2. Pengguna (*user*)
   a. Google Chrome dan Mozila Firefox sebagai browser.

3.2.2 Analisis Kebutuhan Perangkat Keras

Perangkat keras adalah seluruh komponen atau unsur peralatan yang digunakan untuk menunjang pembangunan aplikasi. Adapun perangkat keras yang digunakan untuk pemakaian aplikasi ini secara optimal memerlukan spesifikasi minimum komputer sebagai berikut :

1. Admin dan Pengembang
   a. Processor dengan kecepatan minimal 1.6 GHz+ Motherboard.
   b. Kapasitas Hardisk minimal 160 GB.
   c. VGACard minimal 64 MB.
   d. RAM minimal 2 GB.
   e. Monitor.

f. Mouse.
g. Keyboard.
h. Lan Card 10/100 Mbps.
i. Koneksi internet minimal 64 Kbps.

2. Pengguna (*user*)
   a. Processor dengan kecepatan minimal 1.6 GHz+ Motherboard.
   b. Kapasitas Hardisk minimal 80 GB.
   c. VGACard minimal 64 MB.
   d. RAM minimal 1 GB.
   e. Monitor.
   f. Mouse.
   g. Keyboard.
   h. LAN card.
   i. Koneksi internet minimal 64 Kbps.

3.2.3 Analisis Pengguna

Untuk menjalankan aplikasi ini dibutuhkan spesifikasi *user* yang dapat menjalankan aplikasi ini, baik dalam sisi pengguna maupun admin. Analisis pengguna aplikasi ini memiliki karakteristik antara lain :

1. Admin
   a. Mampu mengunakan komputer dengan baik.
   b. Mampu mengoperasikan internet.
   c. Memiliki hak akses terhadap keseluruhan aplikasi.
   d. Mengetahui pengelolaan data (tambah, ubah, hapus data).
   e. Mengetahui pengelolaan *brainfile.*

2. Pengguna (*user*)
   a. Mampu menggunakan komputer
   b. Mampu mengoperasikan internet
   c. Dapat mengoperasikan aplikasi
   d. User hanya menginputkan teks untuk berinteraksi dengan chatbot

Spesifikasi pengguna diperlukan untuk menjalankan aplikasi. Selain spesifikasi yang telah dicantumkan diatas, batasan umur, serta latar belakang pendidikan tidak mempengaruhi kriteria user untuk menjalankan aplikasi ini. Yang terpenting yaitu *user familiar* dengan penggunaan internet.

3.3 Analisis Kebutuhan Fungsional

Mendefinisikan kebutuhan fungsional dan operasional sistem dengan mendefinisikan skenario penggunaan aplikasi klasifikasi. Pada aplikasi ini yang akan bertindak sebagai aktor yaitu admin dan pengguna. Admin memiliki hak akses penuh, yang mengelola seluruh content perangkat lunak. Pengguna mempunyai hak akses dalam konsultasi dan *interaksi* terhadap dokter *virtual*.

Kebutuhan fungsional dalam pembangunan aplikasi ini mencakup beberapa diagram, dimana diagram-diagram tersebut masing-masing menggambarkan proses-proses yang berjalan pada sistem *chatbot* dokter *virtual*. Tahapan pemodelan dalam analisis tersebut antara lain mengidentifikasi aktor, pembuatan use case diagram, use case scenario, activity diagram, sequence diagram, dan class diagram.

3.3.1 Pemodelan Proses Bisnis

Pemodelan proses bisnis merupakan bagian utama dalam pengembangan arsitektur sistem. Pemodelan arsitektur bisnis yang baik akan menghasilkan kebutuhan arsitektur sistem yang dapat dijadikan dalam dasar pengembangan sistem

Salah satu teknik yang pada saat ini banyak digunakan dalam pengembangan model arsitektur bisnis adalah pendekatan Object Oriented, dimana teknik yang digunakan adalah UML *(Unified Modeling Language*). Secara khusus UML akan digunakan sebagai teknik dalam pemodelan bisnis, dan nantinya akan dijadikan

sebagai dasar dalam pengembangan sistem dengan menitikberatkan pada use case model bisnis dan proses bisnis. Dalam penulisan skripsi ini, pendekatan object oriented digunakan dalam pemodelan proses bisnis *chatbot* dokter *virtual,* bagaimana teknik dan tools yang digunakan dalam *object oriented* untuk menghasilkan model proses bisnis dalam sistem yang akan dibangun.

3.3.2 Identifikasi Aktor

Tahap pertama yang dilakukan dalam melakukan analisis berorientasi objek menggunakan UML adalah menentukan aktor atau pengguna dari sistem. Identifikasi aktor dimaksudkan untuk mengetahui siapa saja aktor yang terlibat di dalam sistem aplikasi ini dokter virtual ini. Deskripsi dari tiap aktor yang terlibat dapat dilihat pada tabel 3.6

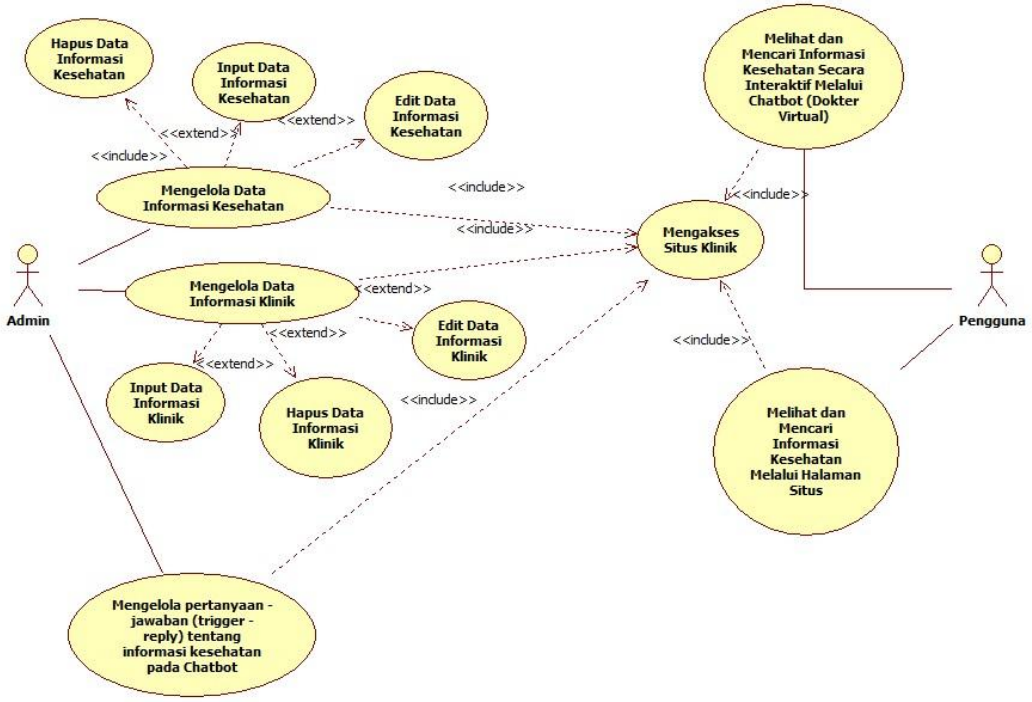
Tabel 3.6 Tabel Identifikasi Aktor

No	Aktor	Deskripsi
1	Admin	Merupakan penggelola sistem *chatbot* dokter *virtual,* Dan penggelola manajemen brainfile pada *chatbot* dokter *virtual*
2	Pengguna	Merupakan pengunjung situs www. kliniksehatbersama.com yang dapat mengakses informasi mengenai klinik dokter iin hartinah dan mencari informasi kesehatan dengan mengunakan sistem *chatbot* dokter *virtual*

3.3.3 Use Case Diagram

Use Case Diagram menggambarkan keterhubungan user dengan sistem, dimana terdiri dari 2 tipe *user*, yaitu admin dan pengguna.

Admin dapat mengakses sistem, yaitu situs klinik dan menggunakannya untuk mengelola data informasi kesehatan, data informasi klinik, dan mengelola pertanyaan-jawaban (*trigger-reply*) oleh *chatbot* dokter *virtual* tentang informasi kesehatan. Pengguna dapat mengakses situs klinik untuk melihat dan mencari informasi kesehatan secara interaktif melalui *Chatbot* dokter *virtual*, dan melihat dan mencari informasi kesehatan melalui halaman situs.



Gambar 3.8 Use Case *Chatbot* Dokter *Virtual*

3.3.3.1 Definisi Use Case *Chatbot* Dokter *Virtual*

Definisi dan deskripsi *Use Case* dapat ditemukan pada table berikut:

Tabel 3.7 Definisi Use Case *Chatbot* Dokter *Virtual*

No	*Use Case*	Deskripsi Singkat
1	Mengelola Data Informasi Kesehatan	Proses yang dilakukan oleh admin untuk menambah / mengurangi informasi mengenai kesehatan secara umum yang akan ditampilkan pada situs klinik.
2	Mengelola Data Informasi Klinik	Proses yang dilakukan oleh admin untuk menambah / mengurangi informasi tentang klinik, seperti jadwal dokter, petunjuk pemeriksaan, biaya, profil dll.
3	Mengelola pertanyaan-jawaban (trigger-reply) tentang informasi kesehatan pada Chatbot	Proses yang dilakukan oleh admin untuk menambah pertanyaan serta jawaban yang akan dilatih untuk Chatbot agar dapat berinteraksi dengan user.
4	Melihat dan mencari informasi kesehatan secara interaktif melalui Chatbot (Dokter Virtual)	Proses yang dilakukan oleh pengguna untuk melihat dan mencari informasi kesehatan yang disajikan secara interaktif melalui proses tanya jawab dengan Chatbot (Dokter Virtual).
5	Melihat dan mencari informasi kesehatan melalui halaman situs	Proses yang dilakukan oleh pengguna untuk melihat dan mencari informasi kesehatan yang disajikan melalui halaman situs dengan cara bernavigasi dan mengakses situs klinik.

3.3.3.2 Skenario Use Case *Chatbot* Dokter *Virtual*

Skenario merupakan penjelasan lebih detail dari kasus (*case*) dari awal hingga akhirnya diperoleh keluaran (*output*).

Tabel 3.8 Deskripsi Use Case Mengelola Data Informasi Kesehatan

Identifikasi		
Nomor	1	
Nama	Mengelola Data Informasi Kesehatan	
Tujuan	Untuk mengelola data informasi kesehatan pada situs klinik	
Deskripsi	Proses yang dilakukan oleh admin untuk mengelola data informasi kesehatan pada situs klinik	
Aktor	Admin	
Skenario Utama		
Kondisi Awal	Mengakses situs klinik	
Aksi Aktor		**Reaksi Sistem**
1. Admin memilih menu pengelolaan data informasi kesehatan pada situ klinik.		
		2. Sistem menampilkan menu input / edit / hapus data informasi kesehatan
Kondisi Akhir	Dokumen / HTML berhasil di*input*kan dan ditampilkan ke layar.	

Tabel 3.9 Deskripsi Use Case Mengelola Data Informasi Klinik

Identifikasi		
Nomor	2	
Nama	Mengelola Data Informasi Klinik	
Tujuan	Untuk mengelola data informasi klinik di situs klinik	
Deskripsi	Proses yang dilakukan oleh admin untuk mengelola data informasi klinik	
Aktor	Admin	
Skenario Utama		
Kondisi Awal	Mengakses situs klinik	
Aksi Aktor		Respon Sistem
1. Admin memilih menu pengelolaan data informasi klinik pada situ klinik.		
		2. Sistem menampilkan menu input / edit / hapus data informasi klinik.
Kondisi Akhir	Dokumen / HTML berhasil di*input*kan dan ditampilkan ke layar.	

Tabel 3.10 Deskripsi Use Case Mengelola Pertanyaan-Jawaban (*Trigger-Reply*) Tentang Informasi Kesehatan Pada *chatbot* dokter *virtual*

Identifikasi		
Nomor	3	
Nama	Mengelola Pertanyaan-Jawaban *(Trigger-Reply)* Tentang Informasi Kesehatan Pada *chatbot* dokter *virtual*	
Tujuan	Untuk mengelola Pertanyaan-Jawaban *(Trigger-Reply)* Tentang Informasi Kesehatan Pada Chatbot	
Deskripsi	Proses yang dilakukan oleh admin untuk mengelola data pertanyaan-jawaban (*trigger-reply)*	
Aktor Utama	Admin	
Skenario Utama		
Kondisi Awal	Mengakses situs klinik	
Aksi Aktor		Respon Sistem
1. Admin memilih menu pengelola pertanyaan-jawaban (*trigger-reply)*		
		2. Sistem menampilkan input dan petunjuk cara memasukkan *trigger-reply*
Kondisi Akhir	Input pertanyaan dan jawaban akan disimpan dalam database dan hasilnya dapat digunakan saat interaksi melalui *chatbot* dokter *virtual*	

Tabel 3.11 Deskripsi Use Case Melihat dan Mencari Informasi Kesehatan Secara Interaktif Melalui *chatbot* dokter *virtual*

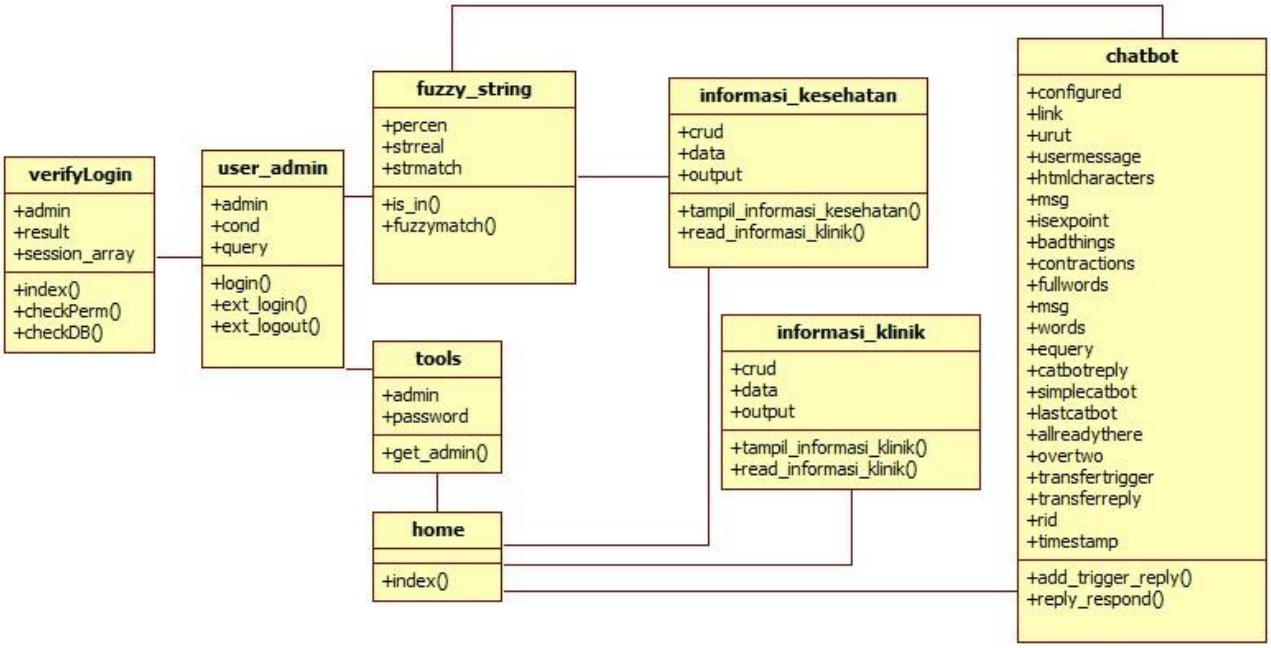
Identifikasi	
Nomor	4
Nama	Melihat dan Mencari Informasi Kesehatan Secara Interaktif Melalui *chatbot* dokter *virtual*
Tujuan	Untuk melihat dan mencari informasi kesehatan secara interaktif melalui chatbot.
Deskripsi	Proses yang dilakukan untuk melihat dan mencari informasi kesehatan secara interaktif melalui
Aktor	Pengguna
Skenario Utama	
Kondisi Awal	Mengakses situs klinik
Aksi Aktor	Respon Sistem
1. Pengguna mengakses halaman chatbot	
	2. Sistem menampilkan halaman *chatbot* dokter *virtual*
Kondisi Akhir	Pertanyaan yang dapat dipahami oleh *chatbot* dokter *virtual* akan dijawab dengan sesuai.

Tabel 3.12 Deskripsi Use Case Melihat dan Mencari Informasi Kesehatan Melalui Halaman Situs

Identifikasi		
Nomor	5	
Nama	Melihat dan Mencari Informasi Kesehatan Melalui Halaman Situs	
Tujuan	Untuk melihat dan mencari informasi kesehatan melalui halaman situs klinik	
Deskripsi	Proses yang dilakukan untuk melihat dan mencari informasi kesehatan melalui halaman situs	
Aktor	*User*	
Skenario Utama		
Kondisi Awal	Mengakses situs klinik.	
Aksi Aktor		Respon Sistem
1 Pengguna mengakses halaman situs dengan informasi yang dicari.		
		3. Sistem menampilkan kolom pencarian infomrasi kesehatan
Kondisi Akhir	Sistem akan menampilkan informasi kesehatan yang dicari oleh pengguna pada layar	

3.3.4 Diagram Kelas

Dalam perancangan diagram kelas untuk keperluan analisis perangkat lunak, maka akan digambarkan kebutuhan yang akan diambil berdasarkan domain permasalahan dan tujuan dari pembuatan perangkat lunak. Perancangan diagram kelas secara keseluruhan akan digambarkan sebagai berikut:



Gambar 3.9 Class Diagram *Chatbot* Dokter *Virtual*

Deskripsi dari setiap kelas diatas dapat digambarkan pada tabel berikut:

Tabel 3.13 Deskripsi Kelas Admin

Deskripsi Kelas		
Nama Kelas	*Admin*	Kelas untuk menangani tipe admin
Property	1. Admin 2. result 3. session_array	1 Atribut tipe admin (true/false) 2 Atribut hasil akses admin (true/false) 3 Attribut data admin (dalam session)
Method	1. index() 2. checkPerm() 3. checkDB()	1. Metode untuk mengakses halaman admin. 2. Metode untuk memeriksa apakah pengguna memiliki hak admin 3. Metode untuk memeriksa koneksi database

Tabel 3.14 Deskripsi Kelas *User* Admin

Deskripsi Kelas		
Nama Kelas	*User Admin*	Kelas untuk menangani proses login
Property	1. Admin 2. cond *3.* query	1. Atribut tipe admin (true / false) 2. Atribut tipe kondisi login 3. Attribut SQL Query
Method	1. login() 2. ext_login() 3. ext_logout()	1. Metode untuk proses login melalui akses ke database 2. Metode untuk login pengguna 3. Metode untuk logout pengguna

Tabel 3.15 Deskripsi Kelas Fuzzy String

Deskripsi Kelas		
Nama Kelas	*Fuzzy String*	Kelas yang menangani proses Fuzzy String
Property	1. Percen 2. strreal 3. strmatch	1. Atribut persentase kecocokan 2. Atribut string awal 3. Atribut string yang cocok
Method	1. is_in() 2. fuzzymatch()	1. Metode untuk mencari apakah ada string yang cocok 2. Metode untuk algoritma fuzzy string

Tabel 3.16 Deskripsi Kelas Tools

Deskripsi Kelas		
Nama Kelas	Tools	Kelas untuk mendapatkan user admin
Property	1. Admin 2. password	1. Attribut untuk mendapatkan user admin 2. Attribut untuk mendapatkan password admin
Method	13. get_admin()	4. Metode untuk mendapatkan user dan password admin

Tabel 3.17 Deskripsi Kelas Informasi Kesehatan

Deskripsi Kelas		
Nama Kelas	*Informasi Kesehatan*	Kelas yang menangani proses CRUD data informasi kesehatan
Property	1.Crud 2.data 3.output	1. Atribut untuk isi dari hasil proses CRUD 2. Atribut untuk menyimpan data indeks 3. Atribut untuk menampilkan isi CRUD di halaman situs
Method	1.*tampil_informasi_kesehatan*() 2. *read_informasi_kesehatan()*	1. Metode untuk menampilkan semua informasi kesehatan 2. Metode untuk membaca data informasi kesehatan

Tabel 3.18 Deskripsi Kelas Informasi Klinik

Deskripsi Kelas		
Nama Kelas	*Informasi Klinik*	Kelas yang menangani proses CRUD data informasi klinik
Property	1. Crud 2. data 3. output	1. Atribut untuk isi dari hasil proses CRUD 2. Atribut untuk menyimpan data indeks 3. Atribut untuk menampilkan isi CRUD di halaman situs
Method	1.*tampil_informasi_kesehatan*() 2. *read_informasi_kesehatan()*	1. Metode untuk menampilkan semua informasi kesehatan 2. Metode untuk membaca data informasi kesehatan

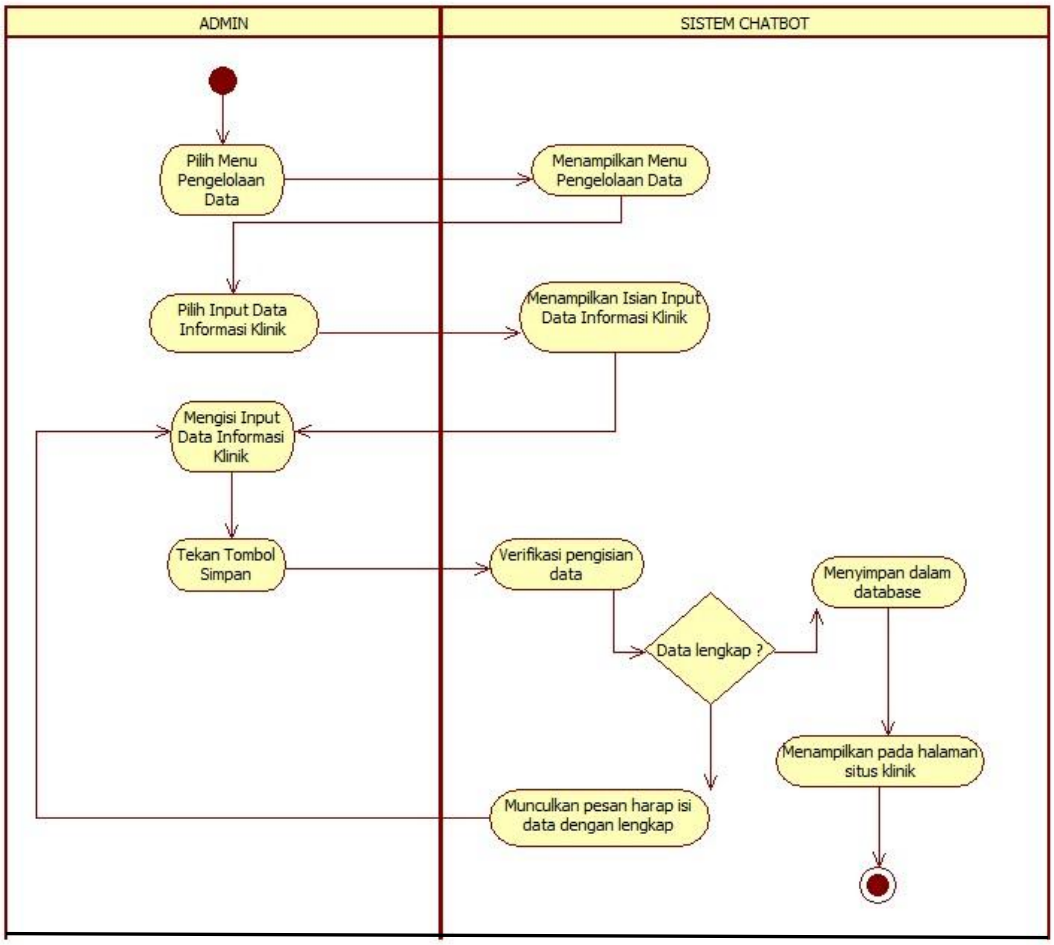
Tabel 3.19 Deskripsi Kelas Chatbot

Deskripsi Kelas		
Nama Kelas	***Chatbot***	Kelas yang digunakan untuk operasional Chatbot
Property	1.*Configured* 2.*link* 3.*urut* 4.*usermessage* 5.*htmlcharacters* 6.*msg* 7.*isexpoint*	1. Atribut untuk kata yang telah diubah 2. Atribut untuk link ke halaman lain 3. Atribut untuk urutan langkah 4. Atribut untuk pesan pengguna 5. Atribut untuk karaketer HTML 6. Atribut untuk

	8.*badthings* 9.*contractions* 10. *fullwords* 11. *msg* 12. *words* 13. *equery* 14. *catbotreply* 15. *simplecatbot* 16. *lastcatbot* 17. *allreadythere* 18. *overtwo* 19. *transfertrigger* 20. *transferreply* 21. *rid* 22. *timestamp*	menampilkan pesan 7. Atribut untuk menyimpan nilai 8. Atribut untuk kata-kata yang salah 9. Atribut untuk kata yang berkontraksi 10. Atribut untuk kata baku 11. Atribut untuk menampilkan pesan 12. Atribut untuk kata 13. Atribut SQL query 14. Atribut untuk reply dari Chatbot 15. Atribut untuk identifikasi Chatbot 16. Atribut untuk pesan akhir 17. Atribut untuk kata yang telah ada 18. Atribut untuk kata yang lebih dari dua 19. Atribut untuk menyimpan trigger 20. Atribut untuk menyimpan reply 21. Atribut untuk ID reply 22. Atribut untuk waktu proses
Method	1. *add_trigger_reply()* 2. *reply_respond()*	1. Metode untuk menyimpan trigger-reply. 2. Metode untuk merespon/reply sesuai trigger

3.3.5 Activity Diagram

3.3.5.1 Activity Diagram Input Data Informasi Klinik



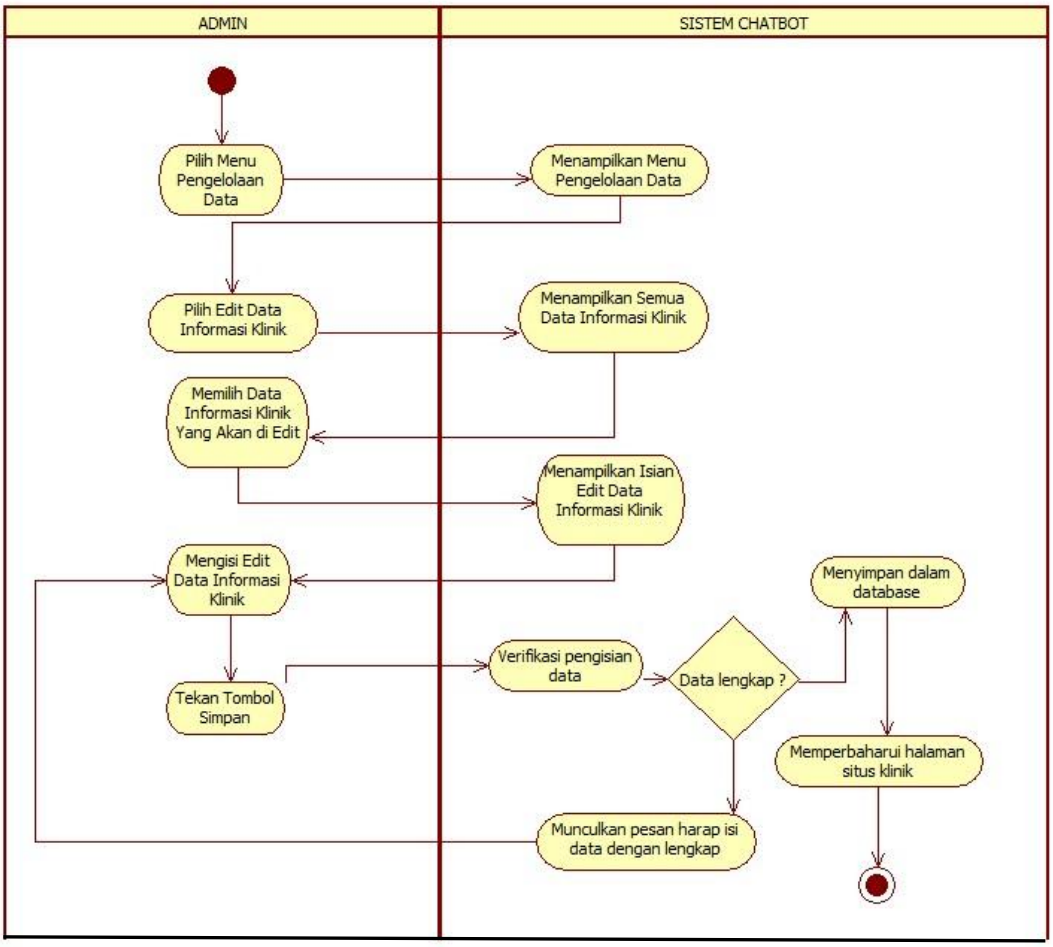
Gambar 3.10 Activity Diagram Input Data Informasi Klinik dr. Iin Hartinah

Pada saat awal Admin mengakses situs klinik, Admin akan memilih menu pengelolaan data, yaitu input data informasi klinik, kemudian Sistem *chatbot* akan menampilkan *form* isian input data informasi klinik tersebut.

Admin akan mengisi input data yang dibutuhkan pada form isian input data, kemudian dilanjutkan dengan menekan tombol simpan. Isian input data tersebut kemudian diverifikasi, jika terjadi kesalahan atau ketidaklengkapan data yang diinput,

maka *chatbot* dokter *virtual* akan menampilkan pesan dan kembali ke *form* isian input data. Namun jika input data telah melewati tahap verifikasi, maka akan disimpan dalam database dan akan ditampilkan pada halaman situs klinik.

3.3.5.2 Activity Diagram *Edit* Data Informasi Klinik

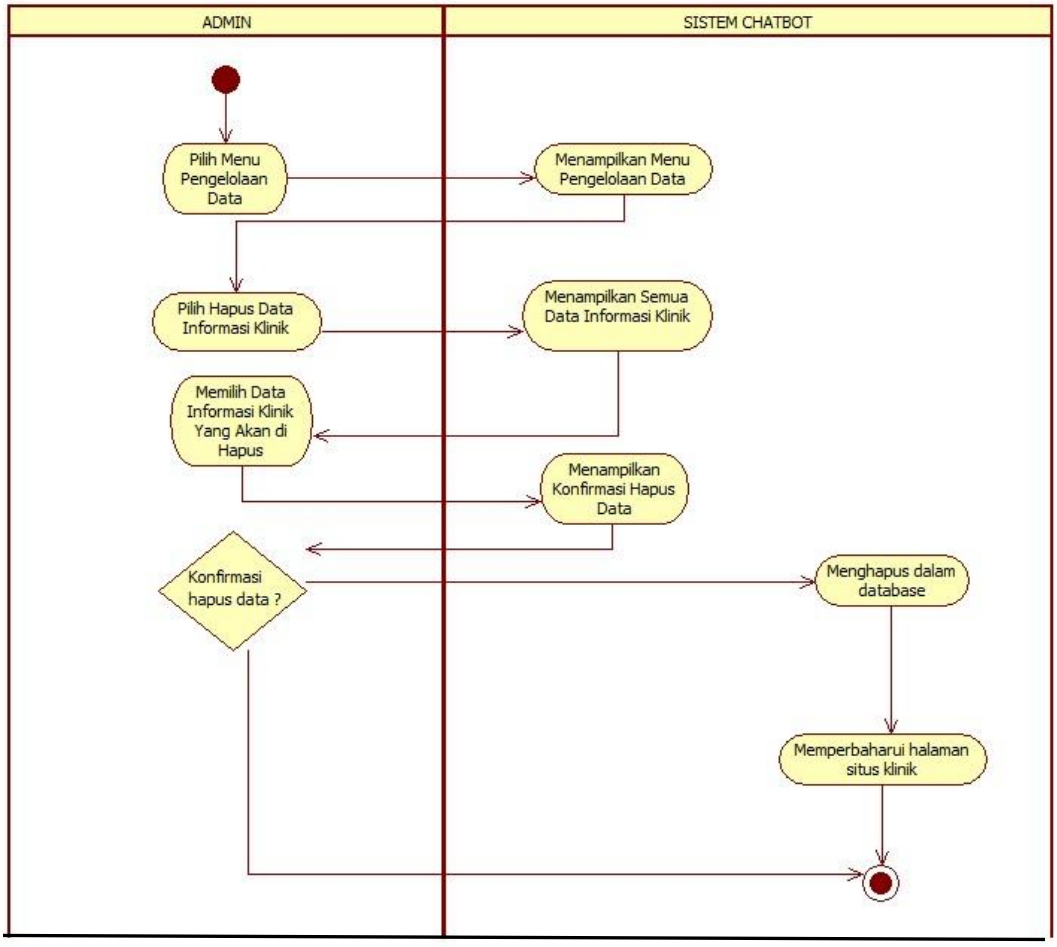


Gambar 3.11 Activity Diagram *Edit* Data Informasi Klinik dr. Iin Hartinah

Pada saat Admin memilih menu pengelolaan data, yaitu edit data informasi klinik, kemudian Sistem *chatbot* dokter *virtual* akan menampilkan semua data informasi yang telah tersimpan. Setelah itu, Admin akan memilih salah satu data informasi yang akan di-*edit* dan kemudian Sistem *chatbot* dokter *virtual* akan menampilkan data informasi yang telah dipilih tersebut.

Admin akan mengisi *edit* data yang dibutuhkan pada form isian edit data, kemudian dilanjutkan dengan menekan tombol simpan. Isian edit data tersebut kemudian diverifikasi, jika terjadi kesalahan atau ketidaklengkapan data yang diedit, maka *chatbot* dokter *virtual* akan menampilkan pesan dan kembali ke form isian edit data. Namun jika *edit* data telah melewati tahap *verifikasi*, maka akan disimpan dalam database dan akan ditampilkan pada halaman situs klinik.

3.3.5.3 Activity Diagram Hapus Data Informasi Klinik



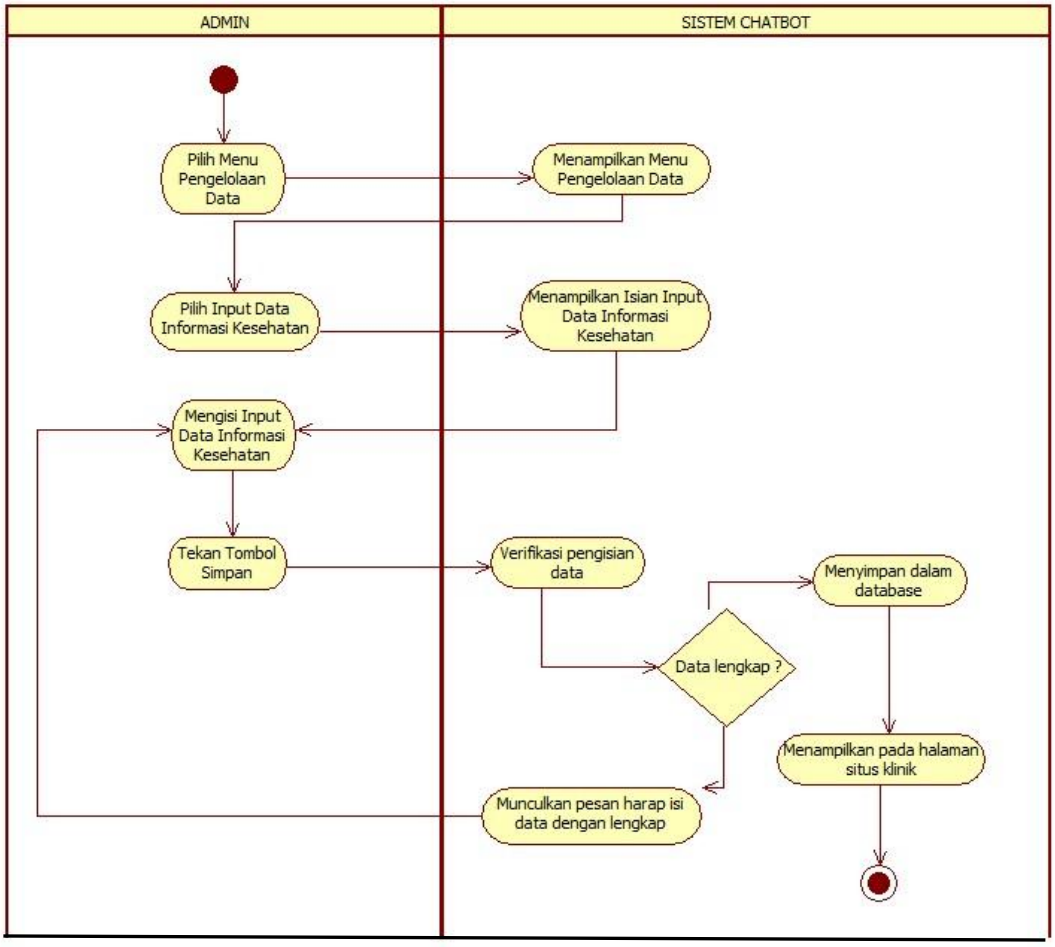
Gambar 3.12 Activity Diagram Hapuss Data Informasi Klinik dr. Iin Hartinah

Pada saat Admin memilih menu pengelolaan data, yaitu hapus data informasi klinik, kemudian Sistem *chatbot* dokter *virtual* akan menampilkan semua data

informasi yang telah tersimpan. Setelah itu, Admin akan memilih salah satu data informasi yang akan dihapus dan kemudian Sistem *chatbot* dokter *virtual* akan menampilkan konfirmasi hapus data.

Admin akan mengkorfimasi hapus data, jika admin memilih untuk menghapus, maka Sistem *chatbot* dokter *virtual* akan menghapus data informasi tersebut dari *database* dan kemudian memperbaharuinya pada halaman situs klinik.

3.3.5.4 Activity Diagram *Input* Data Informasi Kesehatan

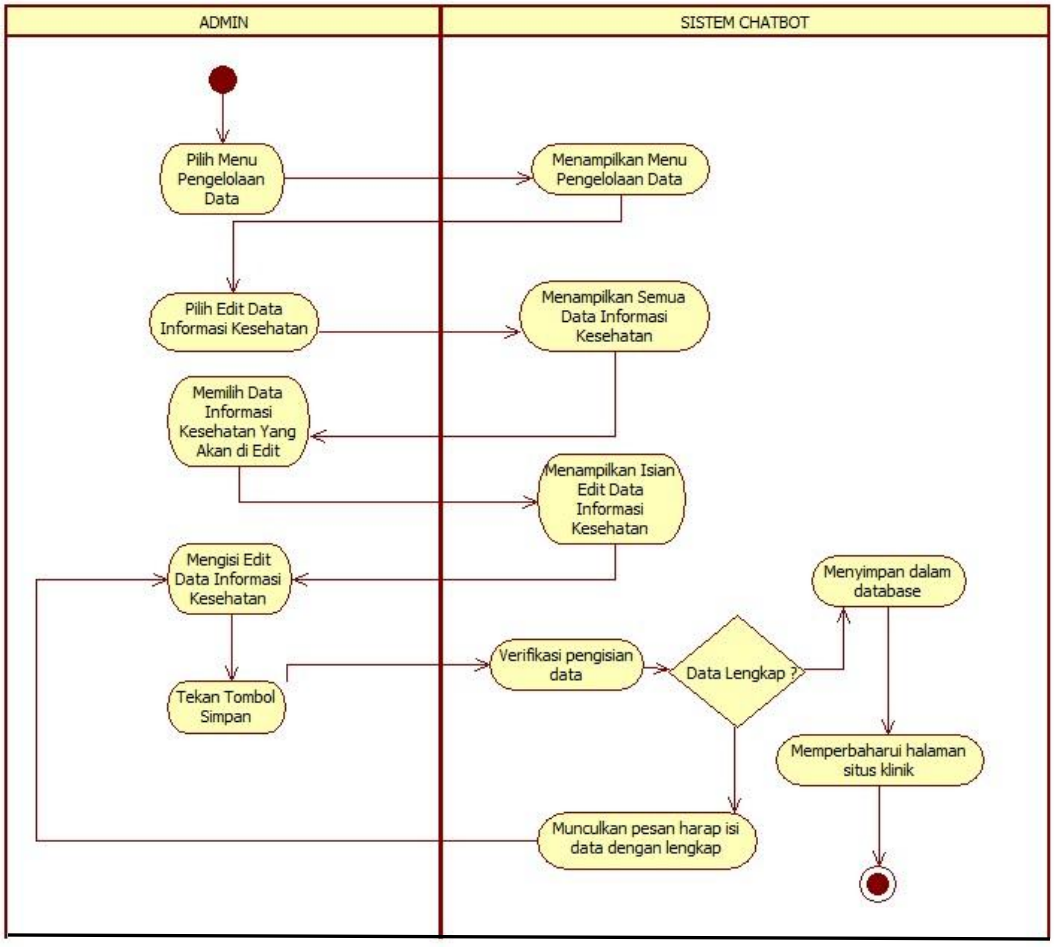


Gambar 3.13 Activity Diagram *Input* Data Informasi Kesehatan

Pada saat awal Admin mengakses situs klinik, Admin akan memilih menu pengelolaan data, yaitu input data informasi kesehatan, kemudian Sistem *chatbot* dokter *virtual* akan menampilkan form isian input data informasi kesehatan tersebut.

Admin akan mengisi input data yang dibutuhkan pada form isian input data, kemudian dilanjutkan dengan menekan tombol simpan. Isian input data tersebut kemudian diverifikasi, jika terjadi kesalahan atau ketidaklengkapan data yang diinput, maka Sistem *chatbot* dokter *virtual* akan menampilkan pesan dan kembali ke form isian input data. Namun jika input data telah melewati tahap verifikasi, maka akan disimpan dalam *database* dan akan ditampilkan pada halaman situs klinik.

3.3.5.5 Activity Diagram Edit Data Informasi Kesehatan

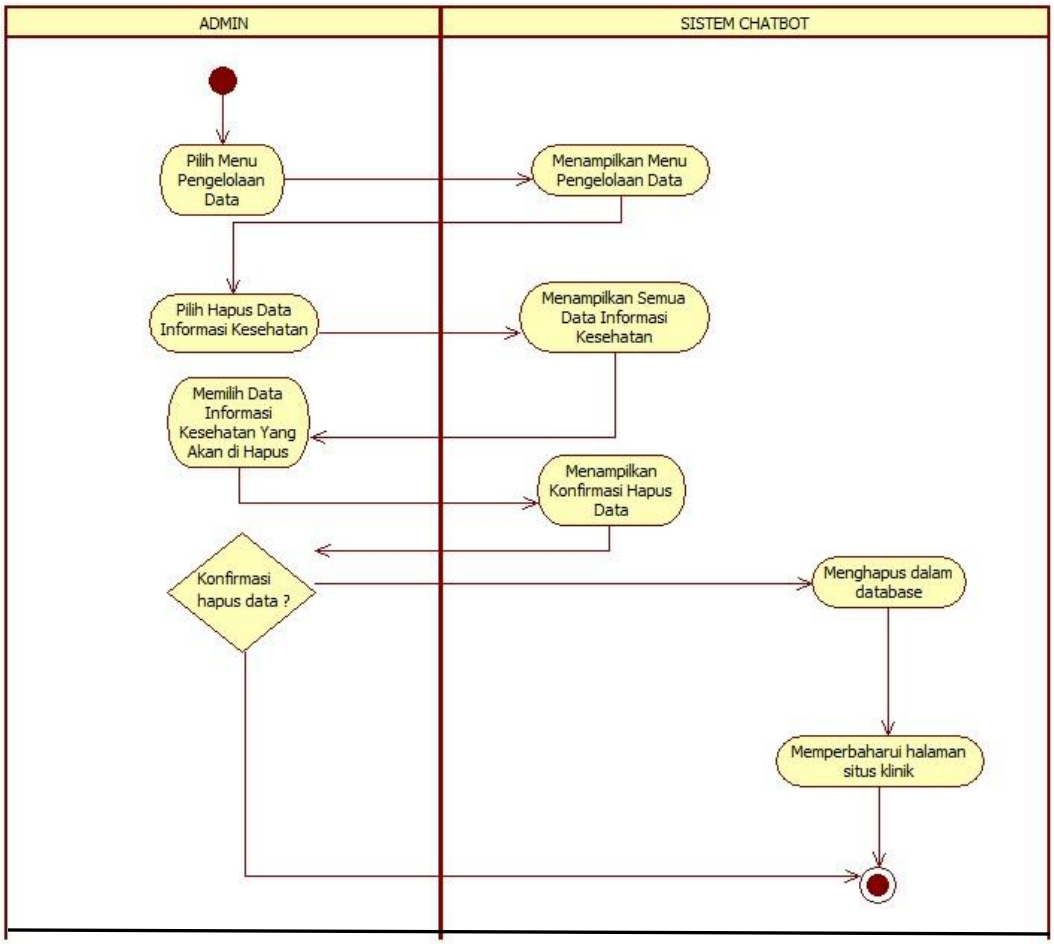


Gambar 3.14 Activity Diagram Edit Data Informasi Kesehatan

Pada saat Admin memilih menu pengelolaan data, yaitu edit data informasi kesehatan, kemudian Sistem *chatbot* dokter *virtual* akan menampilkan semua data informasi yang telah tersimpan. Setelah itu, Admin akan memilih salah satu data informasi yang akan di-edit dan kemudian Sistem Chatbot akan menampilkan data informasi yang telah dipilih tersebut.

Admin akan mengisi edit data yang dibutuhkan pada form isian edit data, kemudian dilanjutkan dengan menekan tombol simpan. Isian edit data tersebut kemudian diverifikasi, jika terjadi kesalahan atau ketidaklengkapan data yang diedit, maka Sistem *chatbot* dokter *virtual* akan menampilkan pesan dan kembali ke form isian edit data. Namun jika edit data telah melewati tahap verifikasi, maka akan disimpan dalam *database* dan akan ditampilkan pada halaman situs klinik.

3.3.5.6 Activity Diagram Hapus Data Informasi Kesehatan

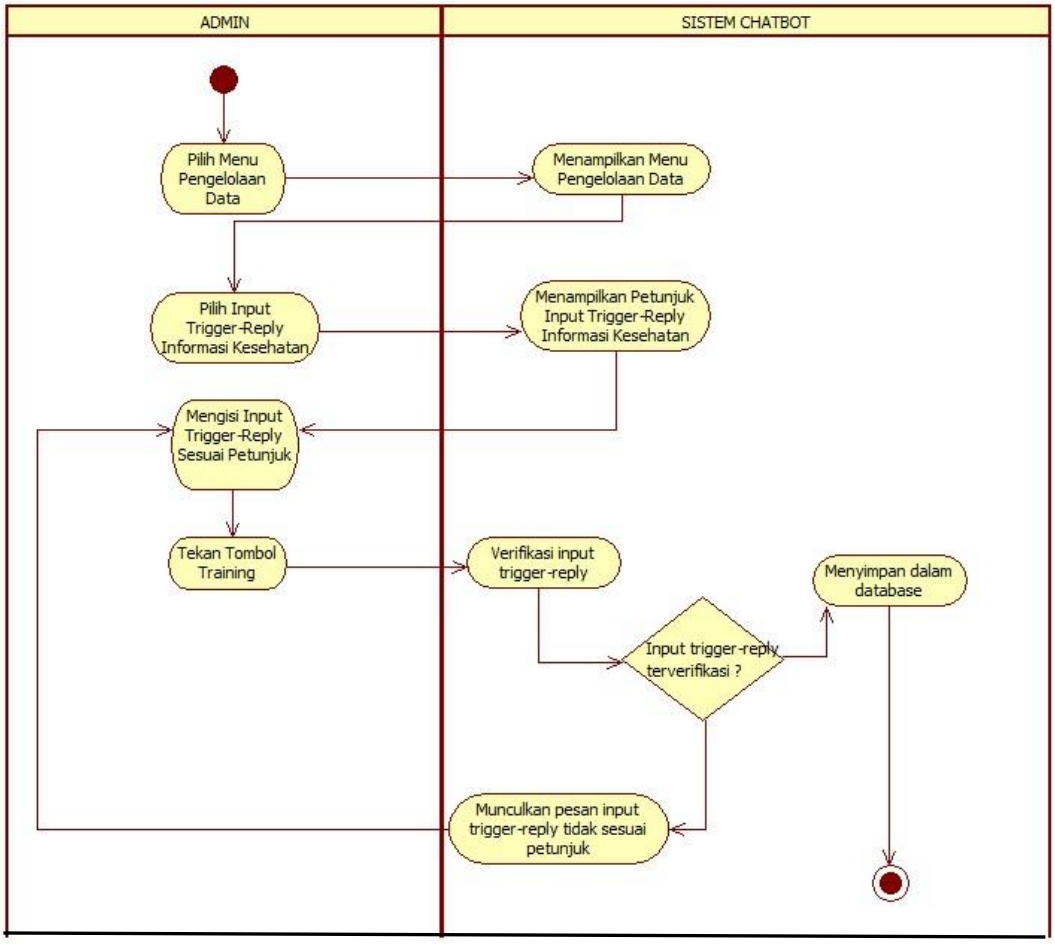


Gambar 3.15 Activity Diagram Hapus Data Informasi Kesehatan

Pada saat Admin memilih menu pengelolaan data, yaitu hapus data informasi kesehatan, kemudian Sistem *chatbot* dokter *virtual* akan menampilkan semua data informasi yang telah tersimpan. Setelah itu, Admin akan memilih salah satu data informasi yang akan dihapus dan kemudian Sistem *chatbot* dokter *virtual* akan menampilkan konfirmasi hapus data.

Admin akan mengkorfimasi hapus data, jika admin memilih untuk menghapus, maka Sistem *chatbot* dokter *virtual* akan menghapus data informasi tersebut dari database dan kemudian memperbaharuinya pada halaman situs klinik.

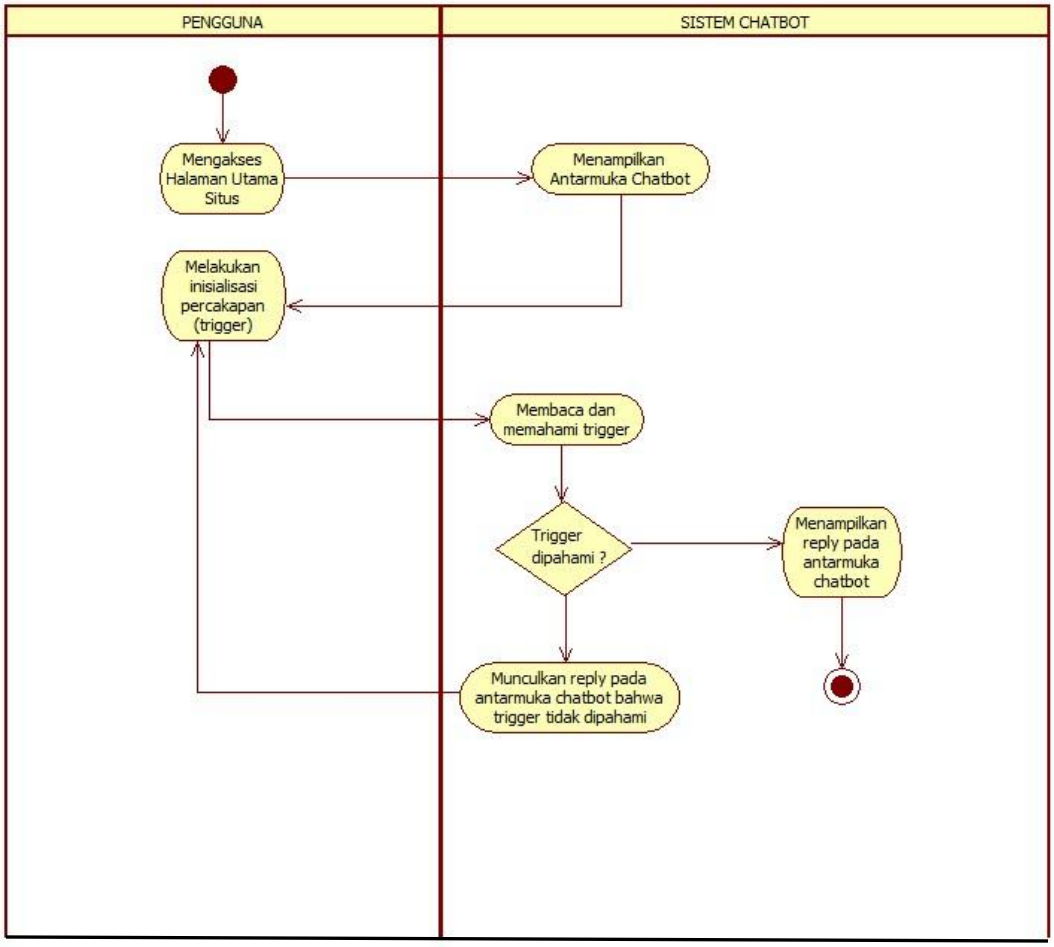
3.3.5.7 Activity Diagram Input Trigger – Reply Informasi Kesehatan Pada Chatbot



Gambar 3.16 Activity Diagram Input Trigger – Reply Informasi Kesehatan Pada Chatbot

Pada saat Admin memilih menu pengelolaan data, yaitu Input Trigger-Reply informasi kesehatan, kemudian Sistem *chatbot* dokter *virtual* akan menampilkan petunjuk input trigger-reply informasi kesehatan. Setelah itu, Admin mengisi input trigger-reply sesuai petunjuk yang diberikan, kemudian menekan tombol training. Input trigger-reply terlebih dahulu diverifikasi apakah telah mengisi sesuai petunjuk. Jika telah sesuai petunjuk maka input tersebut akan disimpan dalam *database*.

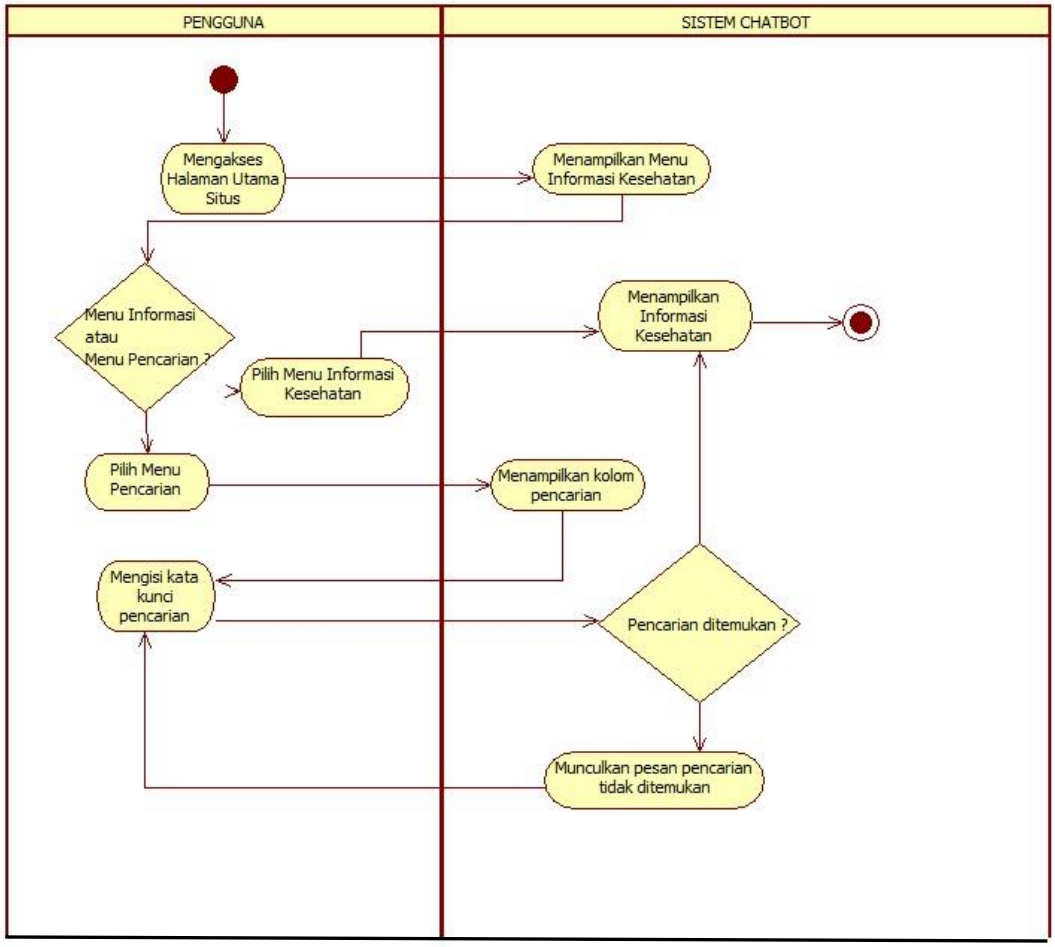
3.3.5.8 Activity Diagram Melihat dan Mencari Informasi Kesehatan Melalui Chatbot



Gambar 3.17 Activity Diagram Melihat dan Mencari Informasi Kesehatan Melalui Chatbot

Pengguna yang mengakses halaman utama situs klinik akan disajikan antarmuka Chatbot oleh sistem. Di antarmuka tersebut, pengguna akan melakukan inisialisasi percakapan (trigger) dengan Chatbot. Percakapan tersebut akan dibaca dan dipahami oleh *chatbot* dokter *virtual* dan jika ada dalam *database* nya, maka *chatbot* dokter *virtual* akan me-respon sesuai dengan isi percakapan (trigger) yang telah diinisialisasi, jika Chatbot tidak memahami percakapan tersebut, maka *chatbot* dokter *virtual* akan menampilkan keterangan bahwa percakapan tersebut tidak dipahaminya.

3.3.5.9 Activity Diagram Melihat dan Mencari Informasi Kesehatan Melalui Halaman Situs

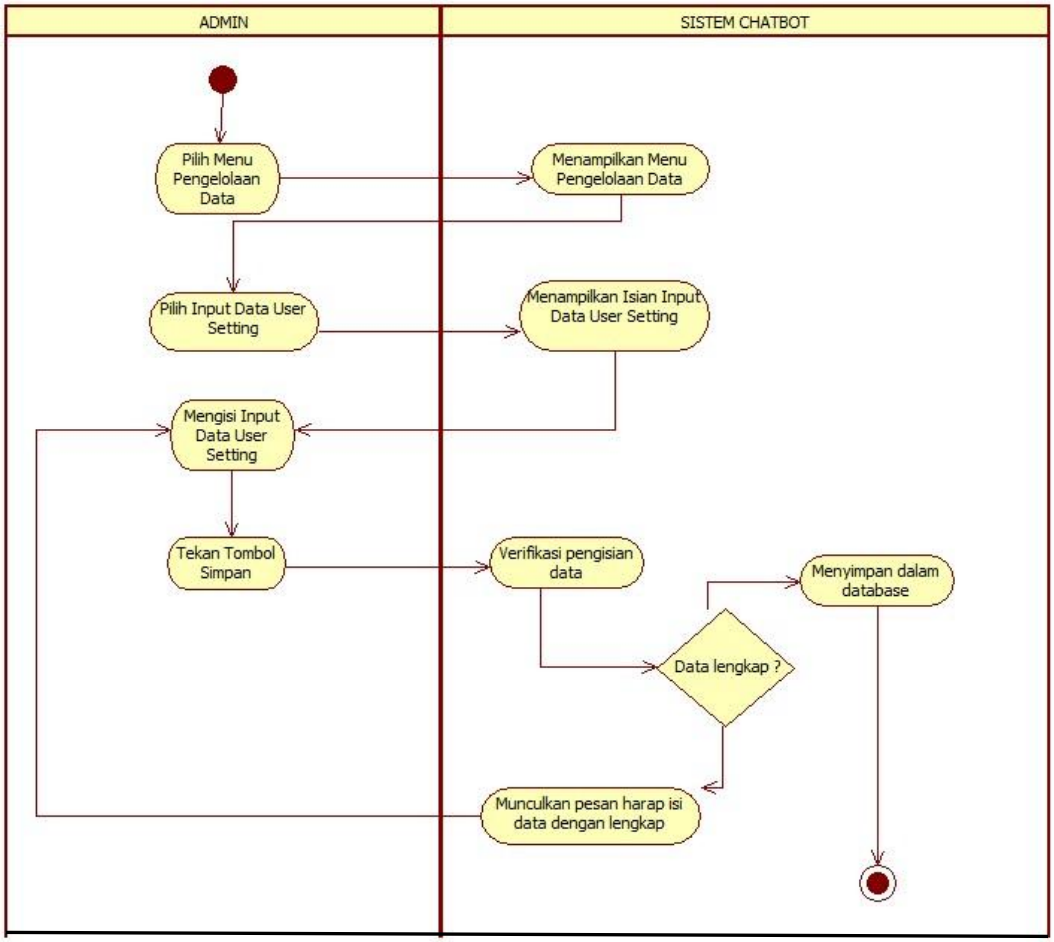


Gambar 3.18 Activity Diagram Melihat dan Mencari Informasi Kesehatan Melalui Halaman Situs

Pengguna yang mengakses halaman utama situs klinik akan disajikan menu informasi kesehatan. Pada menu tersebut terdiri dari menu informasi kesehatan dan menu pencarian. Jika memilih menu informasi kesehatan, maka sistem akan menampilkan informasi sesuai dengan pilihan menu oleh pengguna. Namun jika pengguna memilih menu pencarian, maka sistem akan menampilkan kolom pencarian. Pengguna akan mengisi kata kunci pencarian, jika kata kunci ditemukan maka sistem *chatbot* dokter *virtual* akan menampilkan informasi kesehatan yang

sesuai, jika tidak ditemukan, maka sistem memunculkan pesan informasi bahwa informasi tersebut tidak ditemukan.

3.3.5.10 Activity Diagram *Input* Data *User Setting*



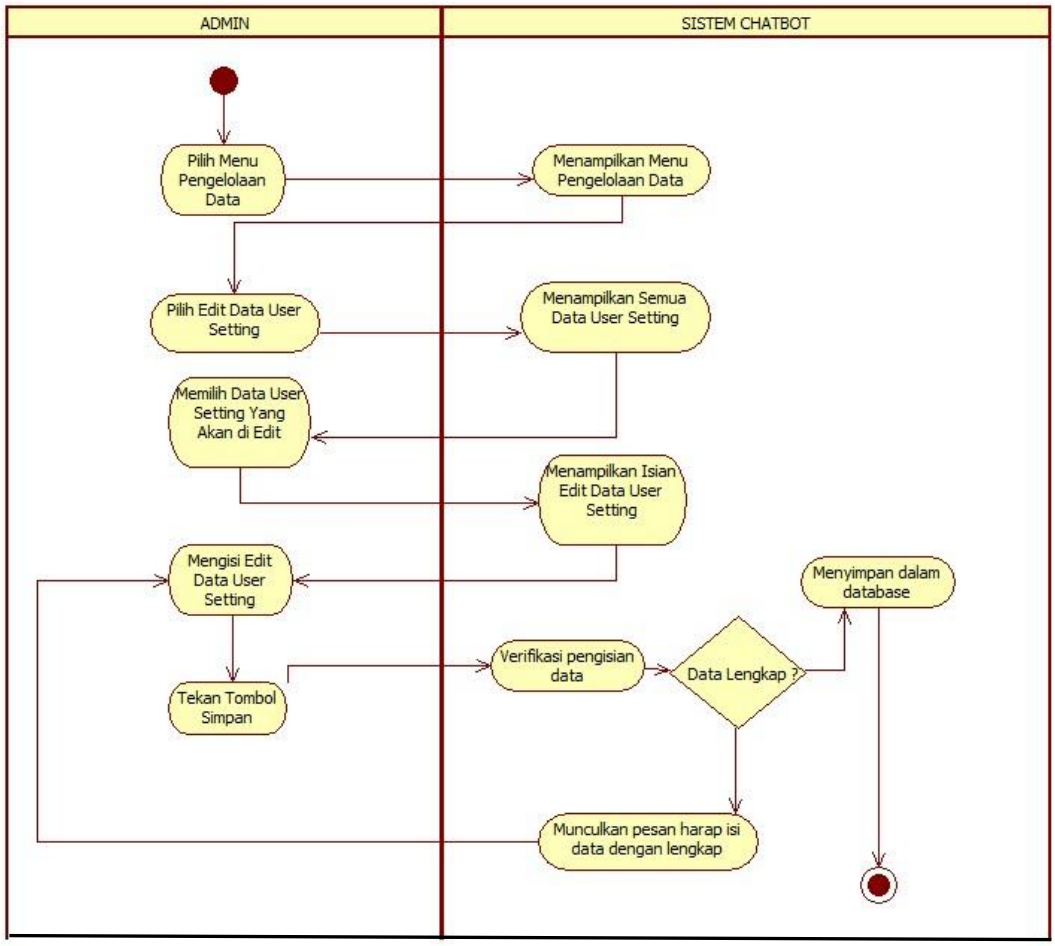
Gambar 3.19 Activity Diagram *Input* Data *User Setting* dr. Iin Hartinah

Pada saat awal Admin mengakses situs klinik, Admin akan memilih menu pengelolaan data, yaitu input data user setting, kemudian Sistem Chatbot akan menampilkan form isian input data user setting tersebut.

Admin akan mengisi input data yang dibutuhkan pada form isian input data, kemudian dilanjutkan dengan menekan tombol simpan. Isian input data tersebut kemudian diverifikasi, jika terjadi kesalahan atau ketidaklengkapan data yang diinput, maka Sistem Chatbot akan menampilkan pesan dan kembali ke form isian input data.

Namun jika input data telah melewati tahap verifikasi, maka akan disimpan dalam database dan akan ditampilkan pada halaman situs klinik.

3.3.5.11 Activity Diagram *Edit* Data *User Setting*

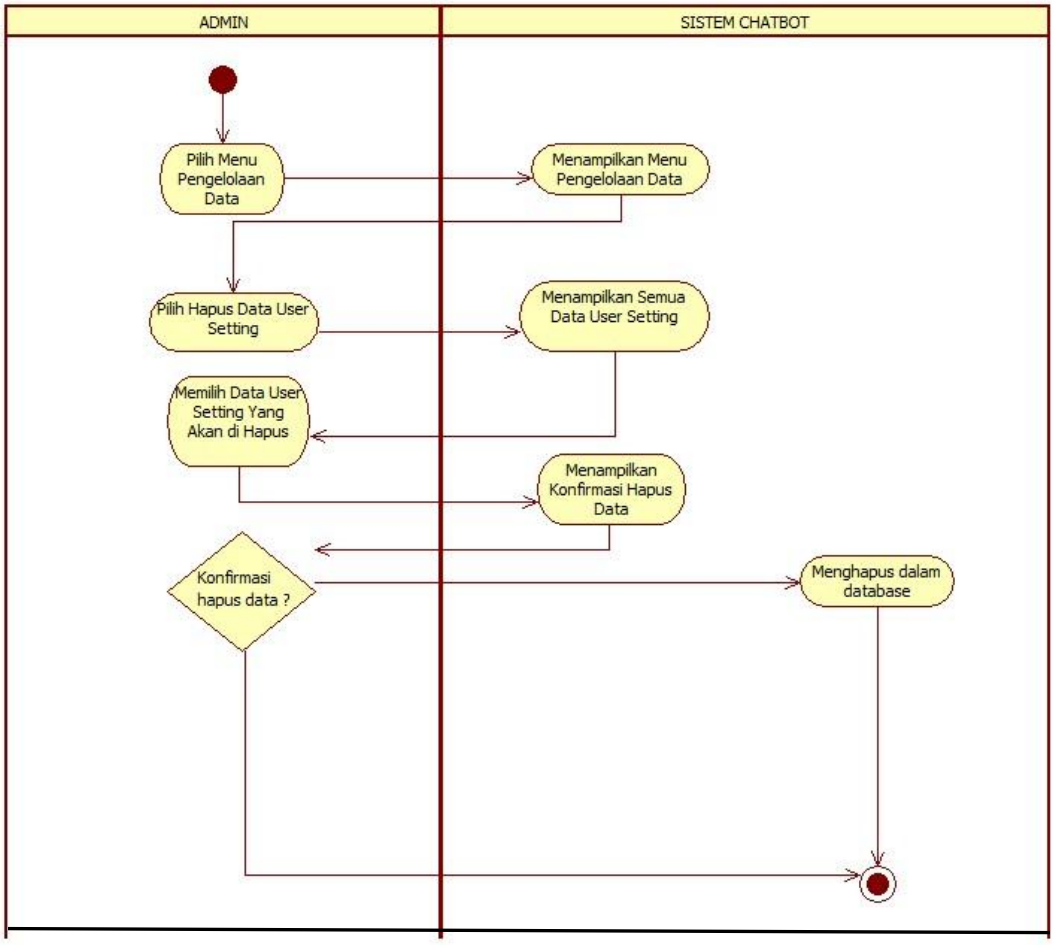


Gambar 3.20 Activity Diagram Edit Data User Setting

Pada saat Admin memilih menu pengelolaan data, yaitu edit data *user setting*, kemudian Sistem *chatbot* dokter *virtual* akan menampilkan semua data informasi yang telah tersimpan. Setelah itu, Admin akan memilih salah satu data informasi yang akan di-edit dan kemudian Sistem *chatbot* dokter *virtual* akan menampilkan data informasi yang telah dipilih tersebut.

Admin akan mengisi edit data yang dibutuhkan pada form isian *edit* data, kemudian dilanjutkan dengan menekan tombol simpan. Isian edit data tersebut kemudian diverifikasi, jika terjadi kesalahan atau ketidaklengkapan data yang diedit, maka Sistem *chatbot* dokter *virtual* akan menampilkan pesan dan kembali ke form isian edit data. Namun jika *edit* data telah melewati tahap verifikasi, maka akan disimpan dalam database dan akan ditampilkan pada halaman situs klinik.

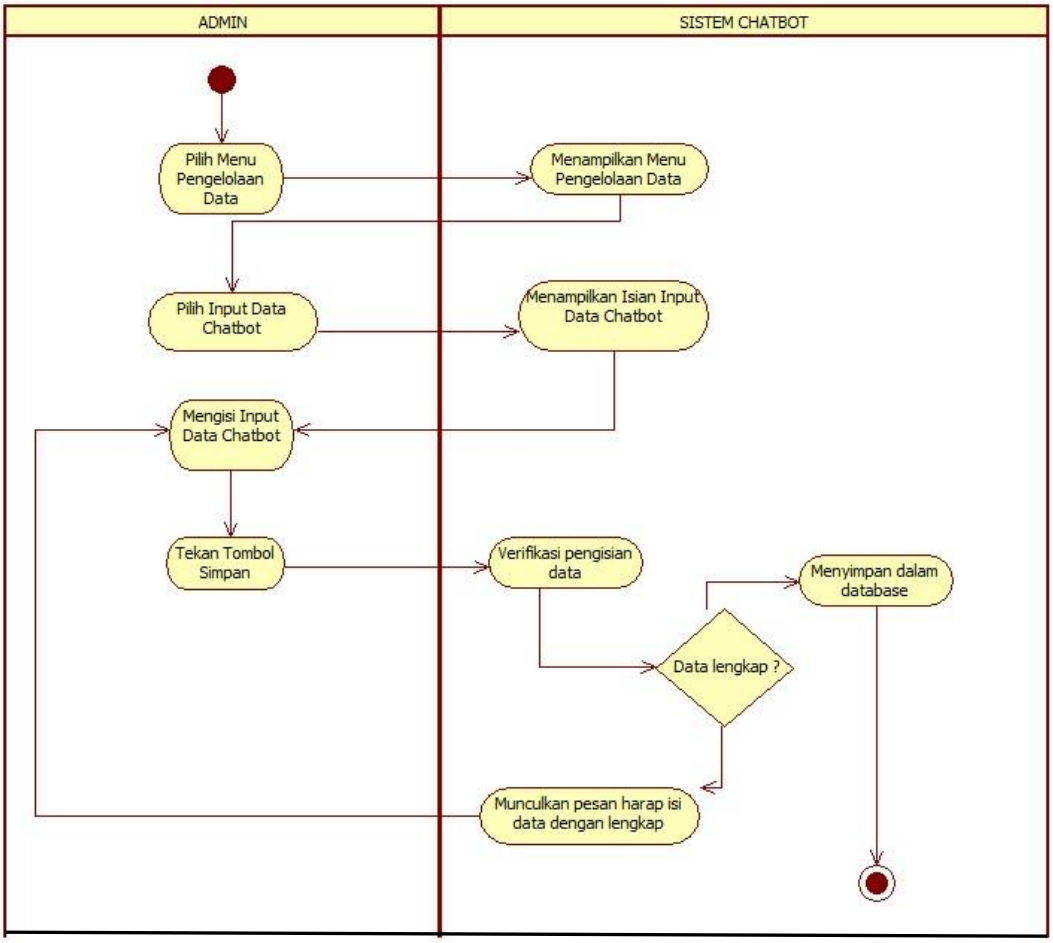
3.3.5.12 Activity Diagram Hapus Data *User Setting*



Gambar 3.21 Activity Diagram Hapus Data *User Setting*

Pada saat Admin memilih menu pengelolaan data, yaitu hapus data *User* Setting, kemudian Sistem *chatbot* dokter *virtual* akan menampilkan semua data informasi yang telah tersimpan. Setelah itu, Admin akan memilih salah satu data informasi yang akan dihapus dan kemudian *chatbot* dokter *virtual* akan menampilkan konfirmasi hapus data.

3.3.5.13 Activity Diagram *Input* Data *Chatbot* (Informasi Kesehatan)

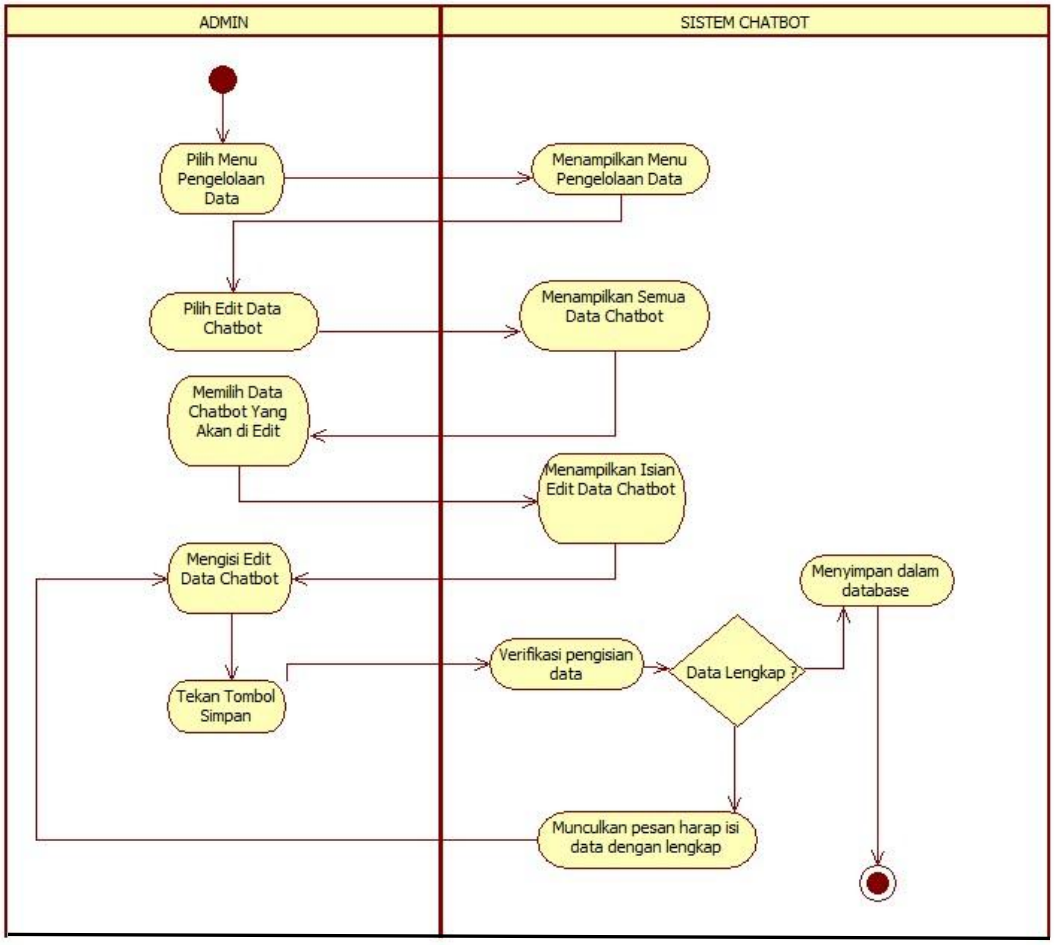


Gambar 3.22 Activity Diagram Input Data Chatbot dr. Iin Hartinah

Pada saat awal Admin mengakses situs klinik, Admin akan memilih menu pengelolaan data, yaitu input data chatbot, kemudian Sistem *chatbot* dokter *virtual* akan menampilkan form isian input data *chatbot* dokter *virtual t* tersebut.

Admin akan mengisi input data yang dibutuhkan pada form isian input data, kemudian dilanjutkan dengan menekan tombol simpan. Isian input data tersebut kemudian diverifikasi, jika terjadi kesalahan atau ketidaklengkapan data yang diinput, maka *chatbot* dokter *virtual* akan menampilkan pesan dan kembali ke form isian input data. Namun jika input data telah melewati tahap verifikasi, maka akan disimpan dalam database dan akan ditampilkan pada halaman situs klinik.

3.3.5.14 Activity Diagram Edit Data Chatbot (Informasi Kesehatan)



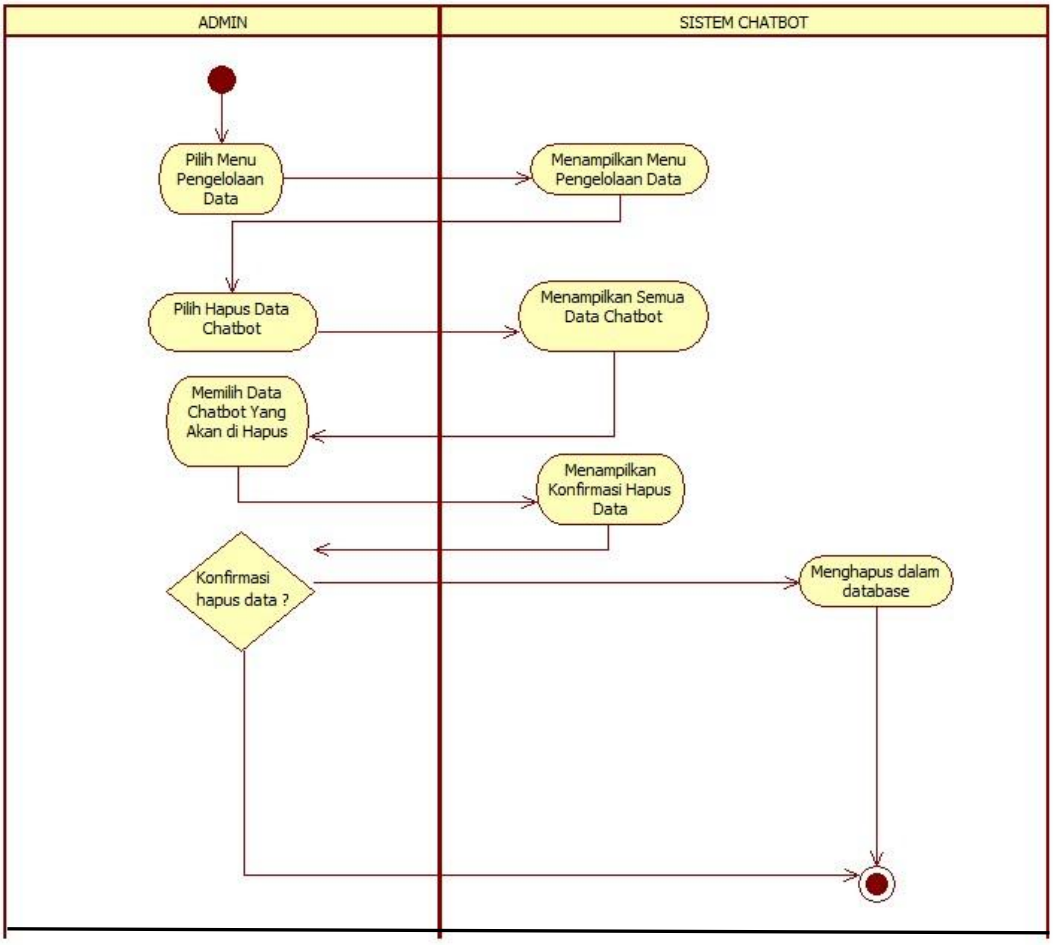
Gambar 3.23 Activity Diagram Edit Data Chatbot

Pada saat Admin memilih menu pengelolaan data, yaitu edit data *chatbot* dokter *virtual* kemudian Sistem *chatbot* dokter *virtual* akan menampilkan semua data

informasi yang telah tersimpan. Setelah itu, Admin akan memilih salah satu data informasi yang akan di-edit dan kemudian Sistem *chatbot* dokter *virtual* akan menampilkan data informasi yang telah dipilih tersebut.

Admin akan mengisi edit data yang dibutuhkan pada form isian edit data, kemudian dilanjutkan dengan menekan tombol simpan. Isian edit data tersebut kemudian diverifikasi, jika terjadi kesalahan atau ketidaklengkapan data yang diedit, maka Sistem Chatbot akan menampilkan pesan dan kembali ke form isian edit data. Namun jika edit data telah melewati tahap verifikasi, maka akan disimpan dalam database dan akan ditampilkan pada halaman situs klinik.

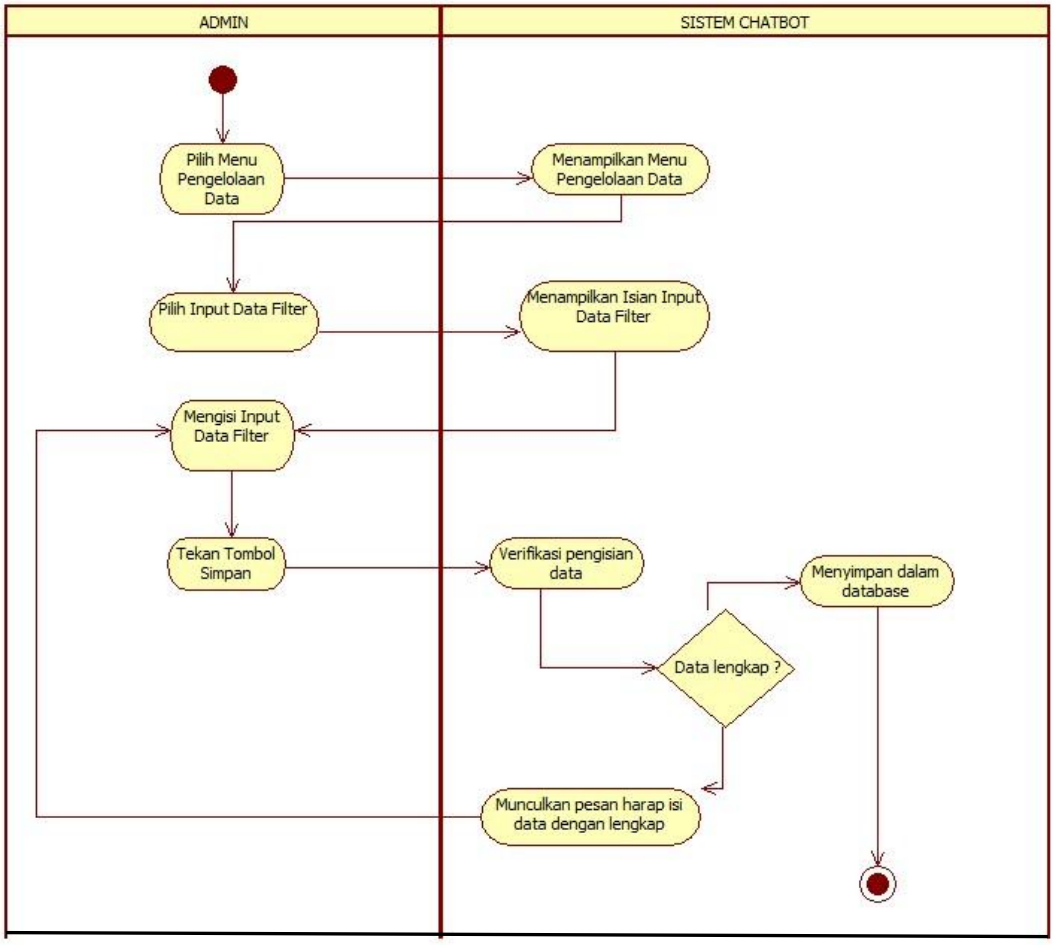
3.3.5.15 Activity Diagram Hapus Data Chatbot (Informasi Kesehatan)



Gambar 3.24 Activity Diagram Hapus Data *chatbot* dokter *virtual* (Informas Kesehatan)

Pada saat Admin memilih menu pengelolaan data, yaitu hapus data *chatbot* dokter *virtual*, kemudian *chatbot* dokter *virtual* akan menampilkan semua data informasi yang telah tersimpan. Setelah itu, Admin akan memilih salah satu data informasi yang akan dihapus dan kemudian Sistem *chatbot* dokter *virtual* akan menampilkan konfirmasi hapus data.

3.3.5.16 Activity Diagram *Input* Data *Filter*

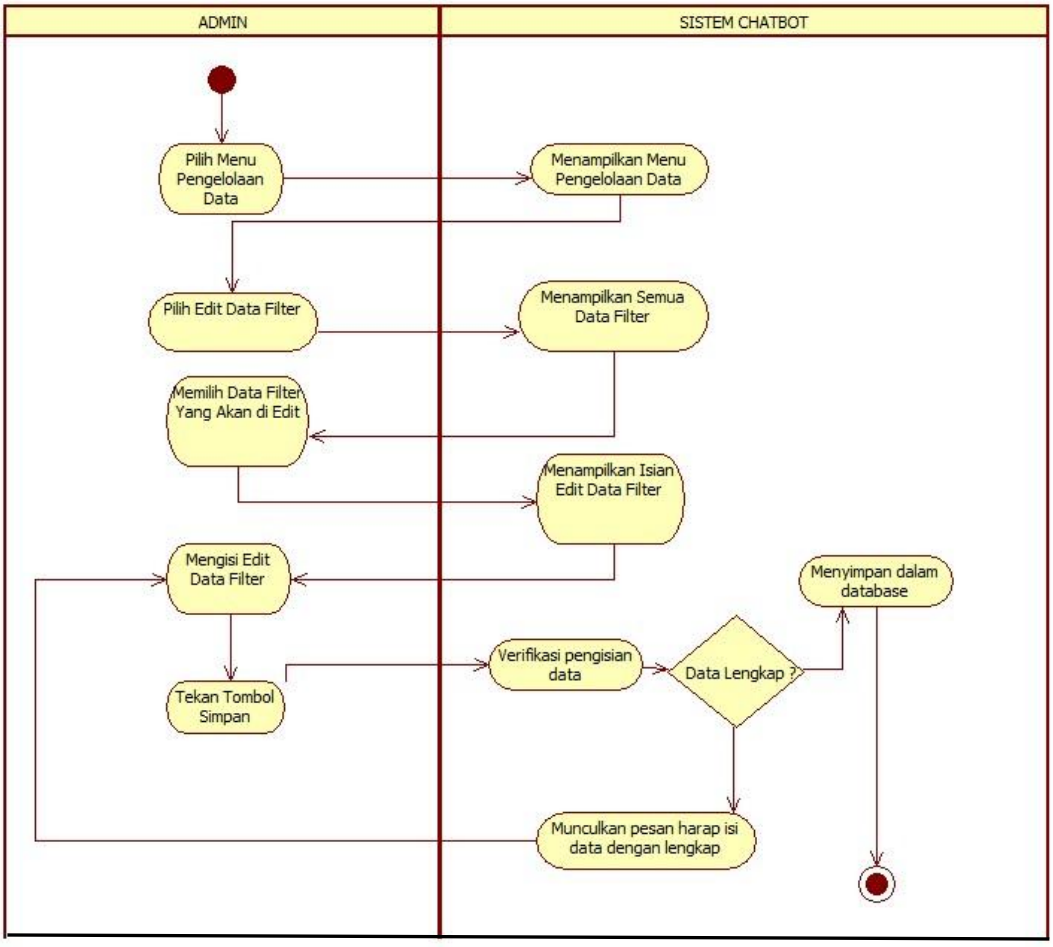


Gambar 3.25 Activity Diagram *Input* Data *Filter* dr. Iin Hartinah

Pada saat awal Admin mengakses situs klinik, Admin akan memilih menu pengelolaan data, yaitu input Data Filter, kemudian *chatbot* dokter *virtual* akan menampilkan form isian *input* Data Filter tersebut.

Admin akan mengisi input data yang dibutuhkan pada form isian input data, kemudian dilanjutkan dengan menekan tombol simpan. Isian input data tersebut kemudian diverifikasi, jika terjadi kesalahan atau ketidaklengkapan data yang diinput, maka *chatbot* dokter *virtual* akan menampilkan pesan dan kembali ke form isian input data. Namun jika *input* data telah melewati tahap verifikasi, maka akan disimpan dalam database dan akan ditampilkan pada halaman situs klinik.

3.3.5.17 Activity Diagram *Edit* Data *Filter*

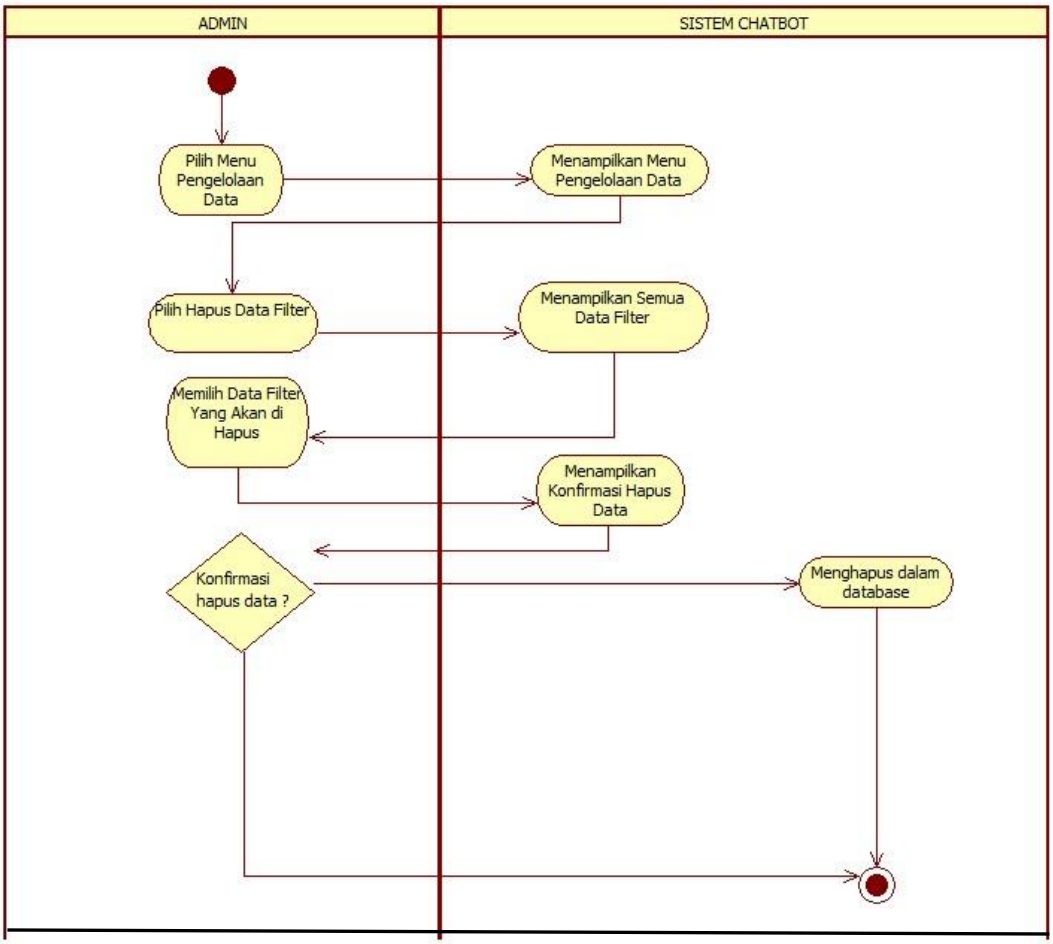


Gambar 3.26 Activity Diagram *Edit* Data Filter

Pada saat Admin memilih menu pengelolaan data, yaitu *edit* Data Filter, kemudian Sistem Chatbot akan menampilkan semua data informasi yang telah tersimpan. Setelah itu, Admin akan memilih salah satu data informasi yang akan di-edit dan kemudian *chatbot* dokter *virtual* akan menampilkan data informasi yang telah dipilih tersebut.

Admin akan mengisi edit data yang dibutuhkan pada form isian edit data, kemudian dilanjutkan dengan menekan tombol simpan. Isian edit data tersebut kemudian diverifikasi, jika terjadi kesalahan atau ketidaklengkapan data yang diedit, maka Sistem Chatbot akan menampilkan pesan dan kembali ke form isian edit data. Namun jika edit data telah melewati tahap verifikasi, maka akan disimpan dalam database dan akan ditampilkan pada halaman situs klinik.

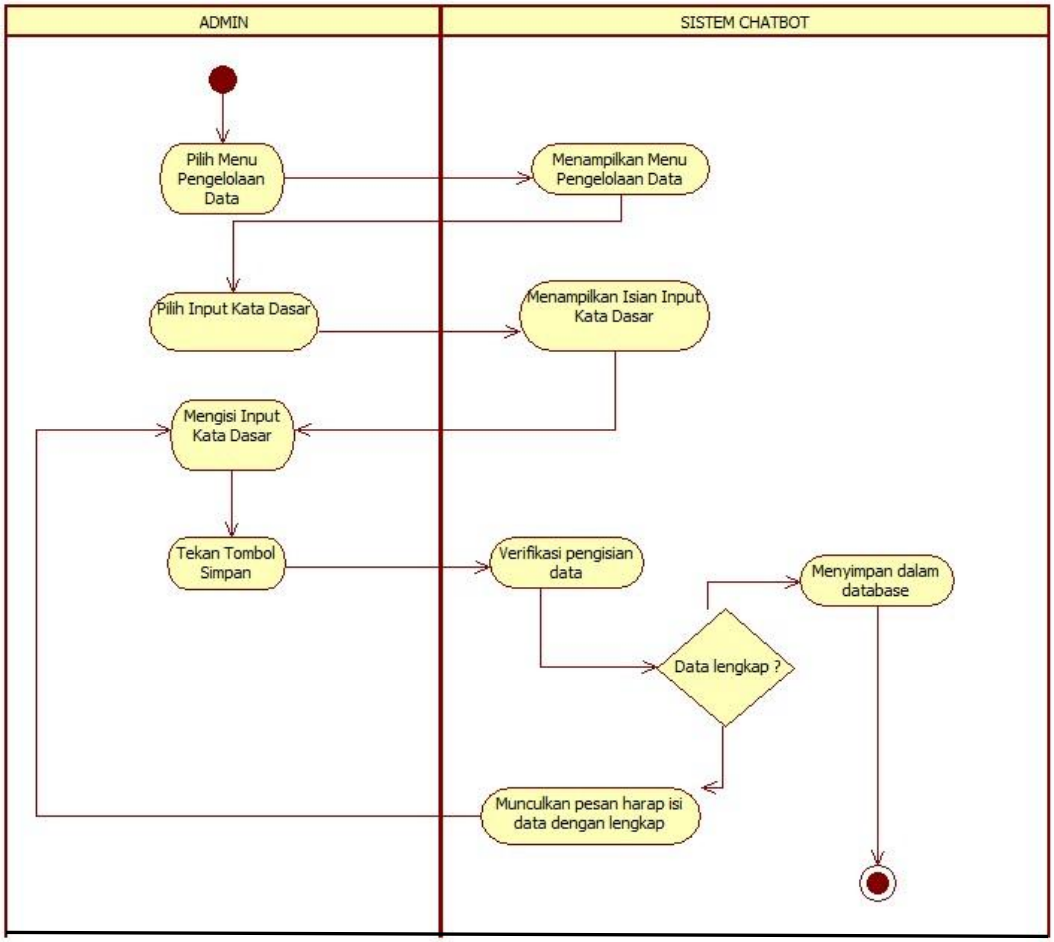
3.3.5.18 Activity Diagram Hapus Data Filter



Gambar 3.27 Activity Diagram Hapus Data *Filter*

Pada saat Admin memilih menu pengelolaan data, yaitu hapus Data Filter, kemudian Sistem *chatbot* dokter *virtual* akan menampilkan semua data informasi yang telah tersimpan. Setelah itu, Admin akan memilih salah satu data informasi yang akan dihapus dan kemudian *chatbot* dokter *virtual* akan menampilkan konfirmasi hapus data.

3.3.5.19 Activity Diagram *Input* Kata Dasar



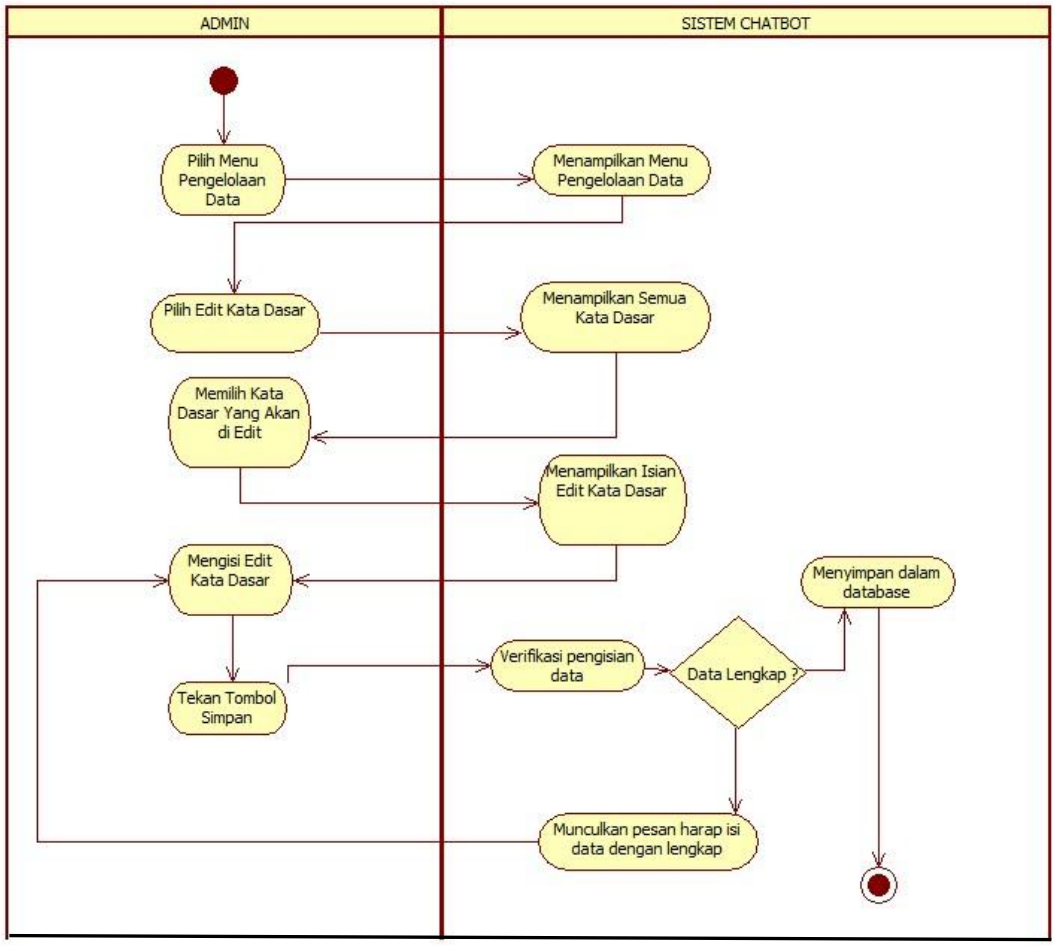
Gambar 3.28 Activity Diagram *Input* Kata Dasar dr. Iin Hartinah

Pada saat awal Admin mengakses situs klinik, Admin akan memilih menu pengelolaan data, yaitu input Kata Dasar, kemudian Sistem *chatbot* dokter *virtual* akan menampilkan form isian input Kata Dasar tersebut.

Admin akan mengisi input data yang dibutuhkan pada form isian input data, kemudian dilanjutkan dengan menekan tombol simpan. Isian *input* data tersebut kemudian diverifikasi, jika terjadi kesalahan atau ketidaklengkapan data yang diinput, maka *chatbot* dokter *virtual* akan menampilkan pesan dan kembali ke form isian

input data. Namun jika input data telah melewati tahap verifikasi, maka akan disimpan dalam database dan akan ditampilkan pada halaman situs klinik.

3.3.5.20 Activity Diagram *Edit* Kata Dasar

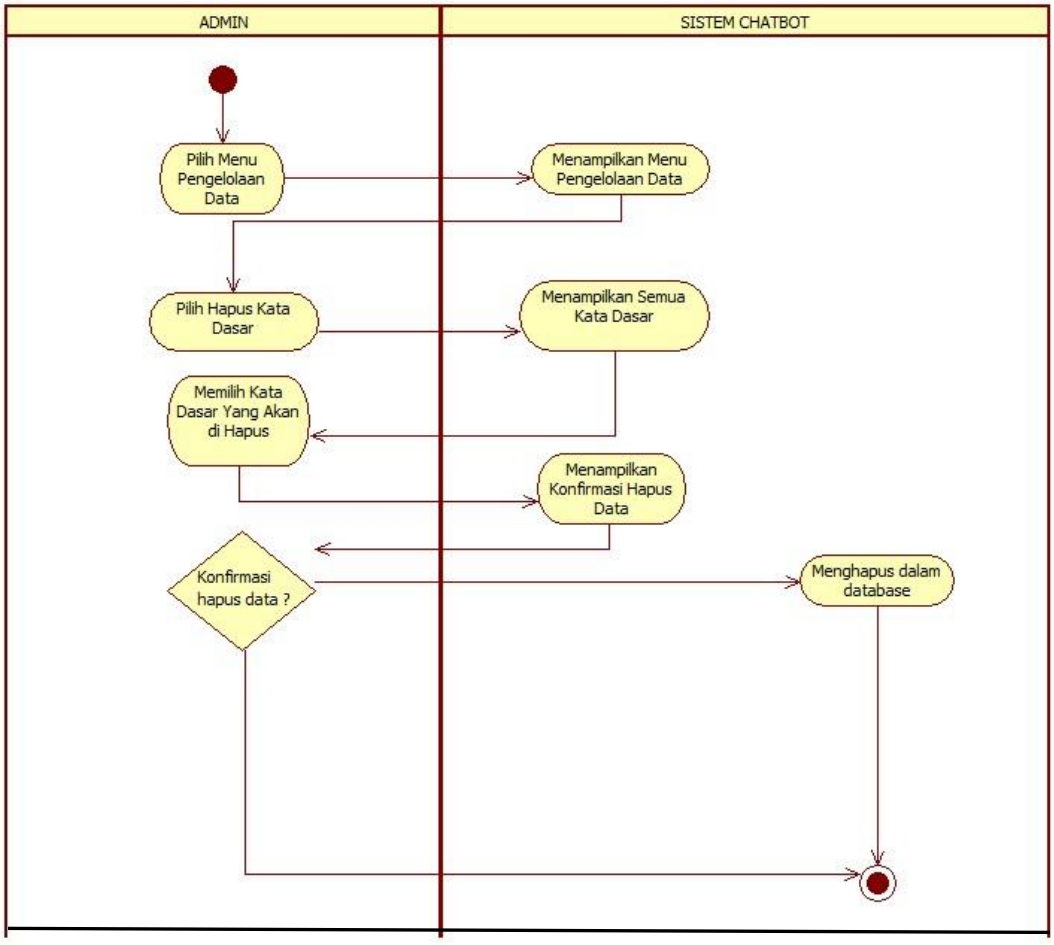


Gambar 3.29 Activity Diagram *Edit* Kata Dasar

Pada saat Admin memilih menu pengelolaan data, yaitu edit Kata Dasar, kemudian Sistem Chatbot akan menampilkan semua data informasi yang telah tersimpan. Setelah itu, Admin akan memilih salah satu data informasi yang akan di-edit dan kemudian Sistem *chatbot* dokter *virtual* akan menampilkan data informasi yang telah dipilih tersebut.

Admin akan mengisi edit data yang dibutuhkan pada form isian edit data, kemudian dilanjutkan dengan menekan tombol simpan. Isian edit data tersebut kemudian diverifikasi, jika terjadi kesalahan atau ketidaklengkapan data yang diedit, maka Sistem *chatbot* dokter *virtual* akan menampilkan pesan dan kembali ke form isian edit data. Namun jika edit data telah melewati tahap verifikasi, maka akan disimpan dalam database dan akan ditampilkan pada halaman situs klinik.

3.3.5.21 Activity Diagram Hapus Kata Dasar

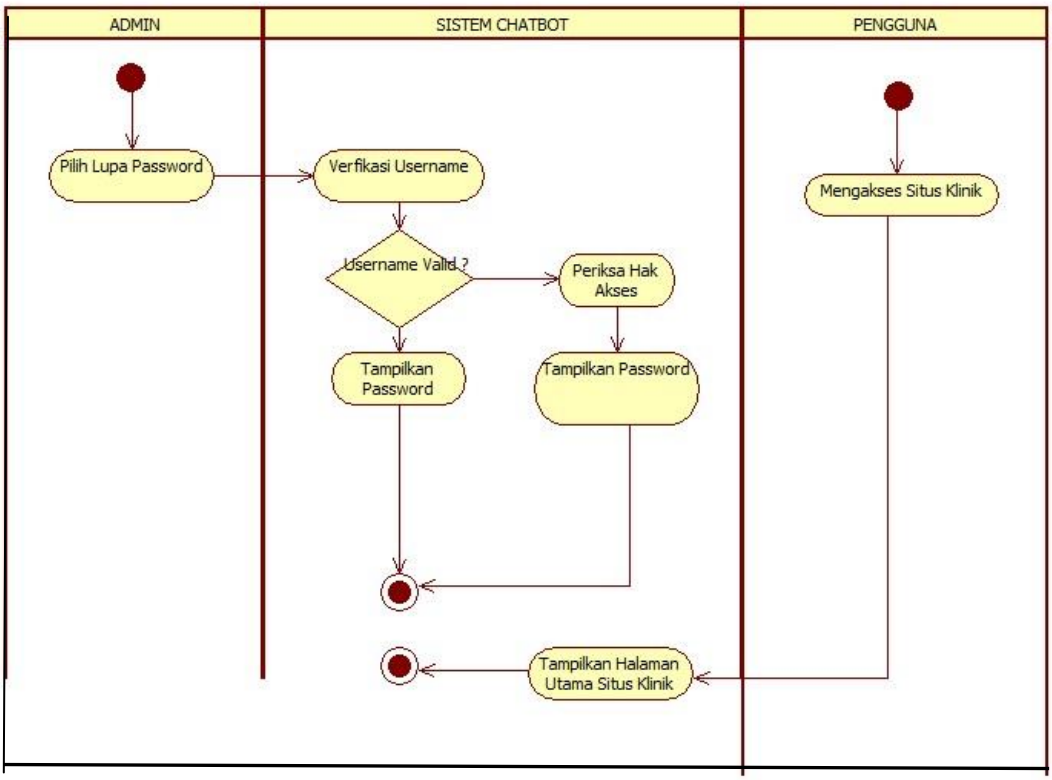


Gambar 3.30 Activity Diagram Hapus Kata Dasar

Pada saat Admin memilih menu pengelolaan data, yaitu hapus Kata Dasar, kemudian Sistem Chatbot akan menampilkan semua data informasi yang telah

tersimpan. Setelah itu, Admin akan memilih salah satu data informasi yang akan dihapus dan kemudian Sistem *chatbot* dokter *virtual* t akan menampilkan konfirmasi hapus data.

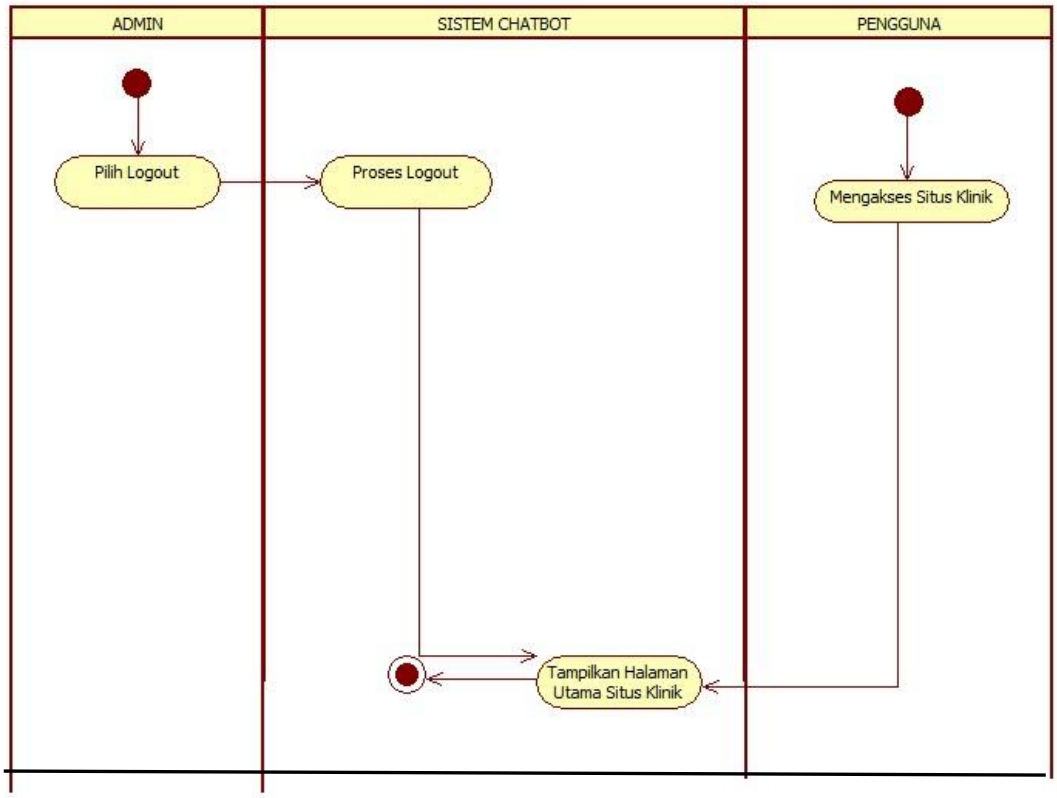
3.3.5.22 Activity Diagram Lupa Password



Gambar 3.31 Activity Diagram Lupa Password

Pada saat Admin atau user lupa password, akan memilih menu lupa password dan memasukkan username-nya, kemudian Sistem Chatbot akan memverifikasi username tersebut jika valid maka Sistem *chatbot* dokter *virtual* akan menampilkan password yang telah tersimpan.

3.3.5.23 Activity Diagram *Logout*



Gambar 3.32 Activity Diagram Logout

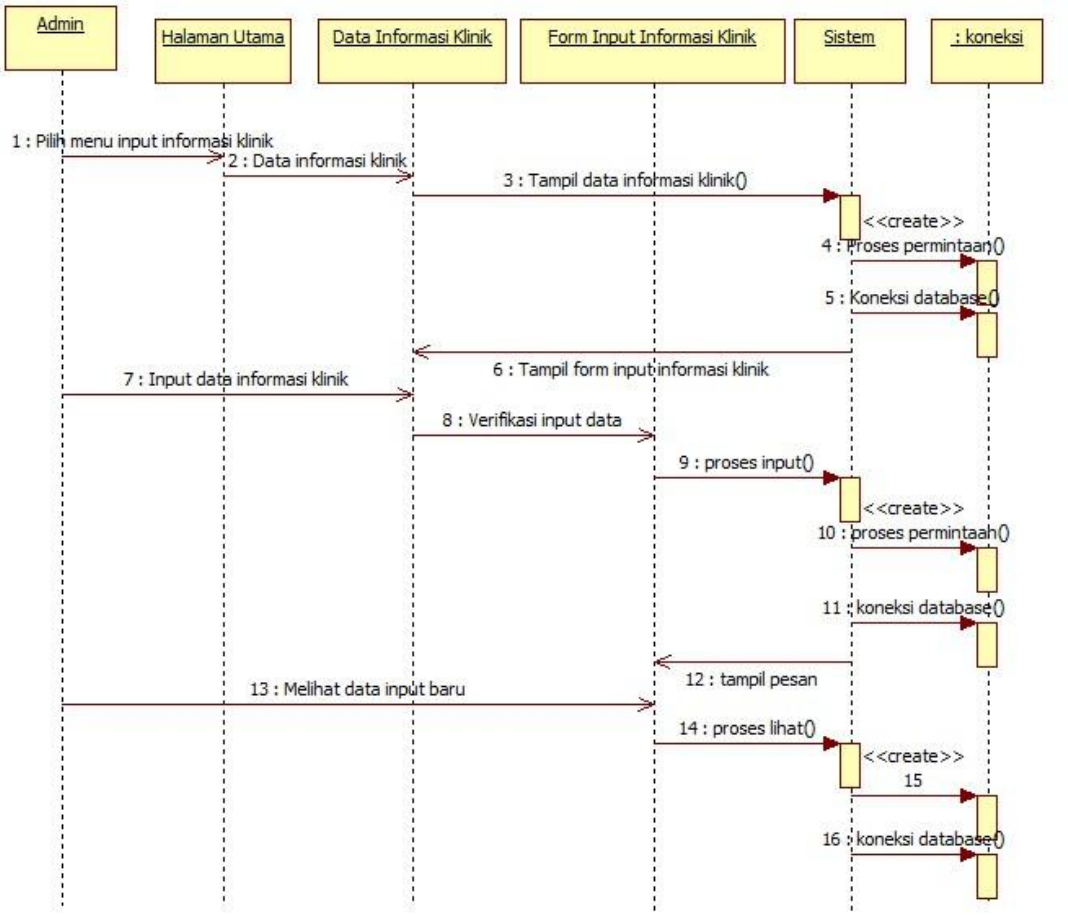
Pada saat Admin atau user melakukan *logout*, akan memilih menu logout, kemudian Sistem *chatbot* dokter *virtual* akan memproses *logout* dan menampilkan kembali halaman utama situs klinik

3.3.6 Perancangan Sequance diagram

Diagram *sequence* menggambarkan alur eksekus program dari use case berupa urutan pemanggilan fungsi dari kelas-kelas yang telah dirancang. Perancangan sequence diagram dapat dilihat pada gambar.dibawah ini.

3.3.6.1 Sequence Diagram *Input* Data Informasi Klinik

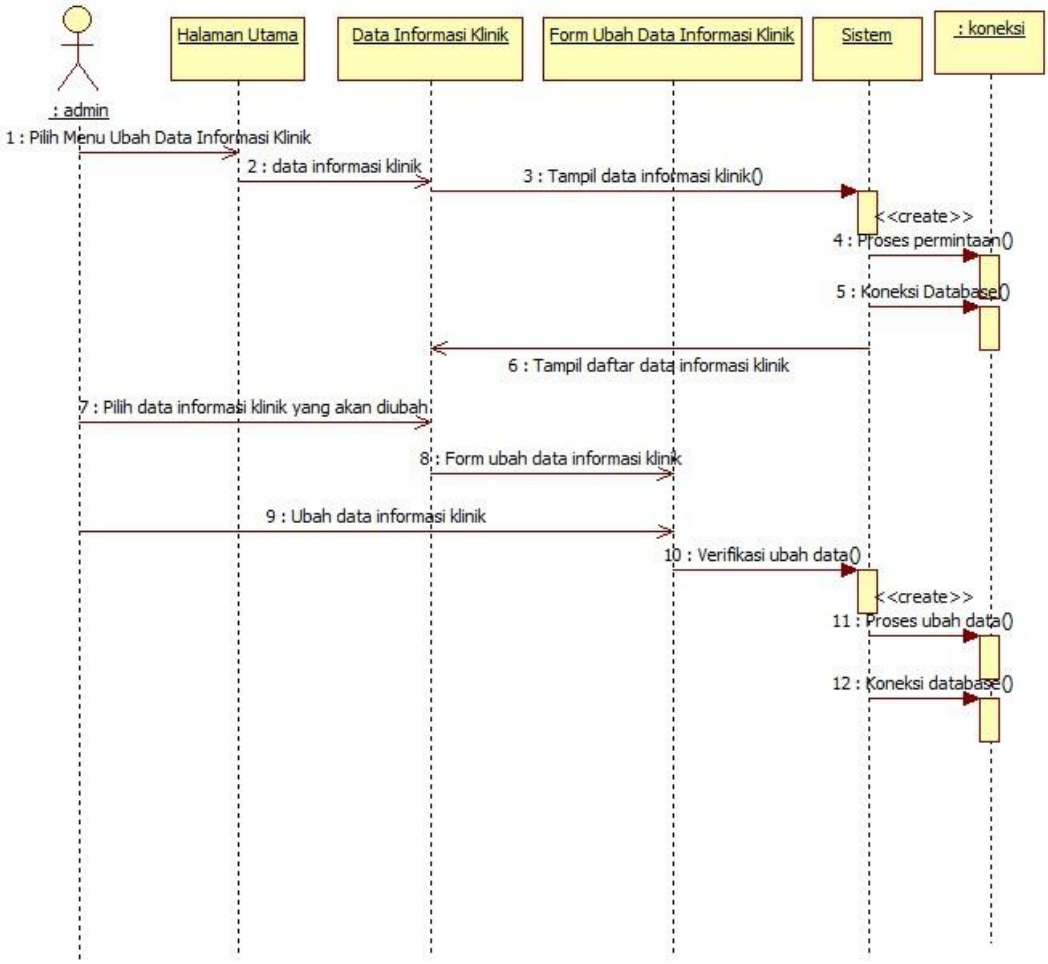
Gambar 3.33 merupakan sequence diagram input data informasi klinik. Proses ini melibatkan admin, aplikasi, serta dokumen html/text yang akan diinputkan kedalam aplikasi untuk diolah lebih lanjut.



Gambar 3.33 Sequence Diagram *Input* Data Informasi Klinik

3.3.6.2 Sequence Diagram *Edit* Data Informasi Klinik

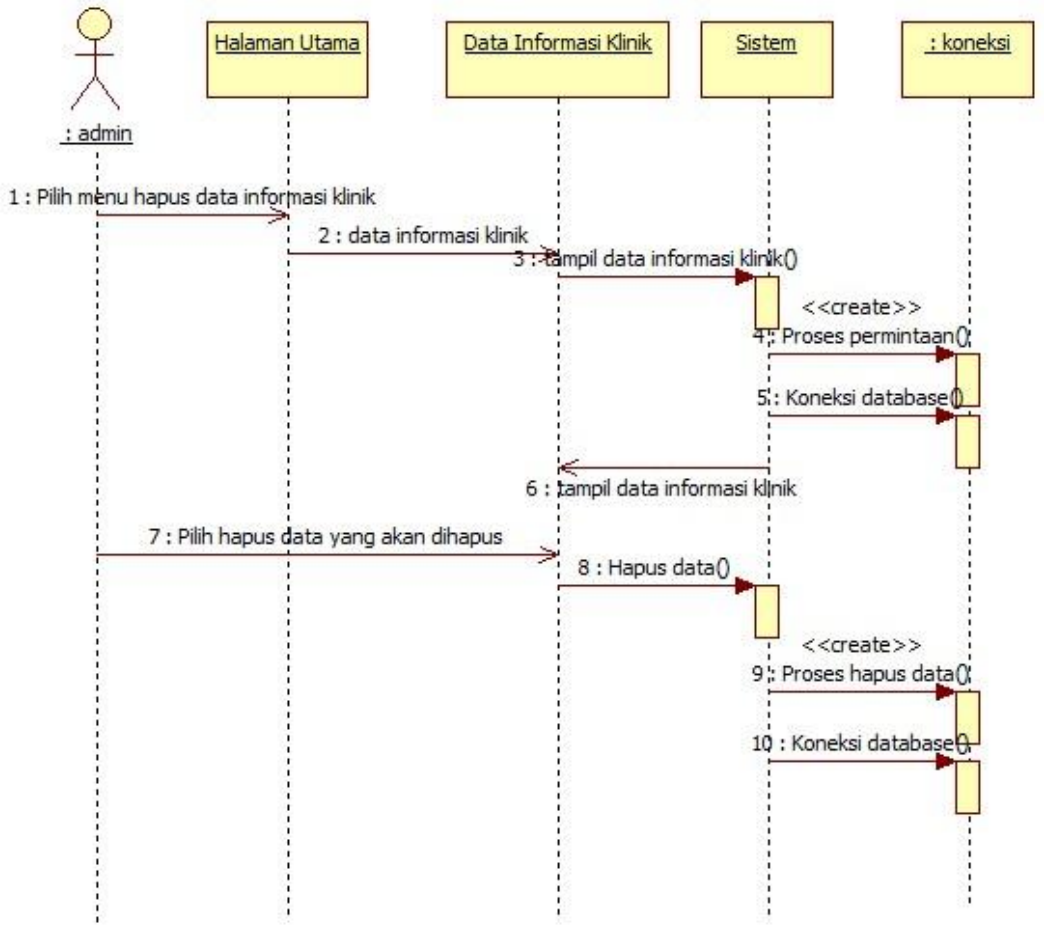
Gambar 3.34 merupakan proses pembaharuan (*edit*) dokumen html/text yang berisi tentang data informasi klinik, yang mana dalam prosesnya melibatkan admin, aplikasi dan dokumen html/text.



Gambar 3.34 Sequence Diagram *Edit* Data Informasi Klinik

3.3.6.3 Sequence Diagram Hapus Data Informasi Klinik

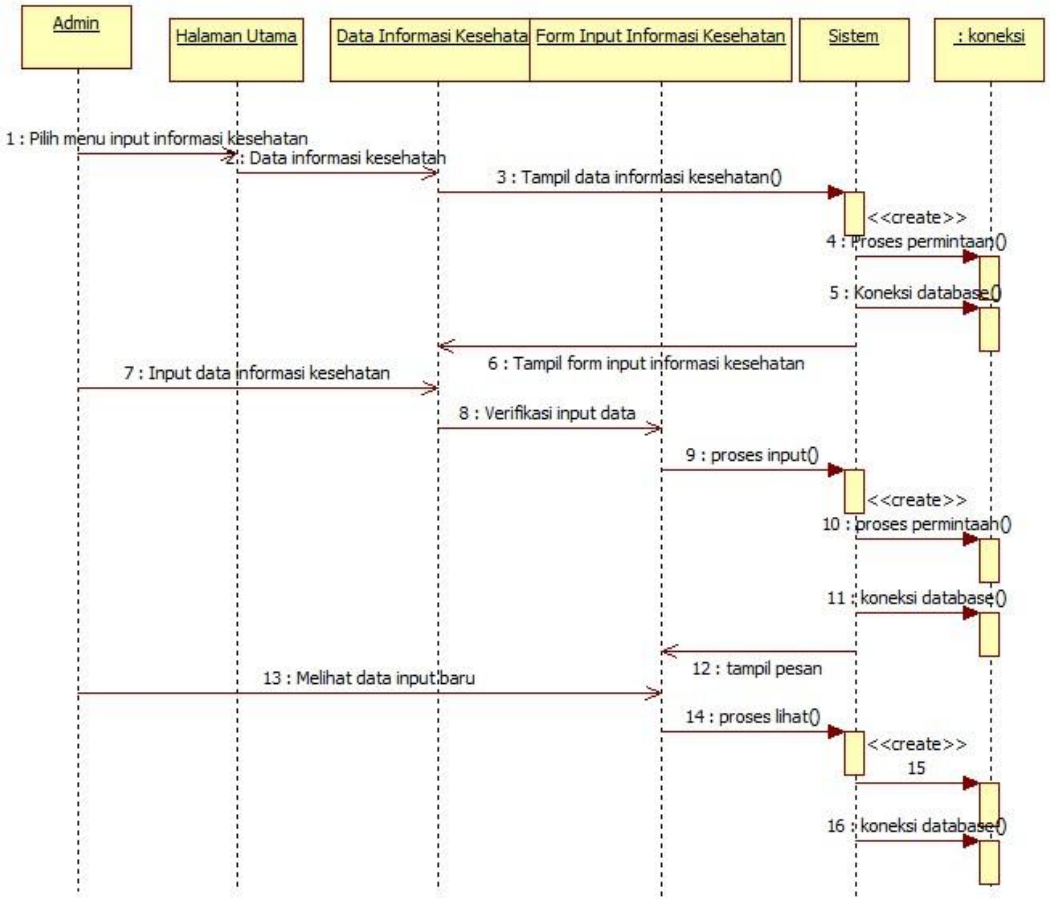
Gambar 3.35 merupakan proses penghapusan data informasi klinik, yang mana dalam prosesnya melibatkan admin, aplikasi, dan dokumen html/text.



Gambar 3.35 Sequence Diagram Hapus Data Informasi Klinik

3.3.6.4 Sequence Diagram *Input* Data Informasi Kesehatan

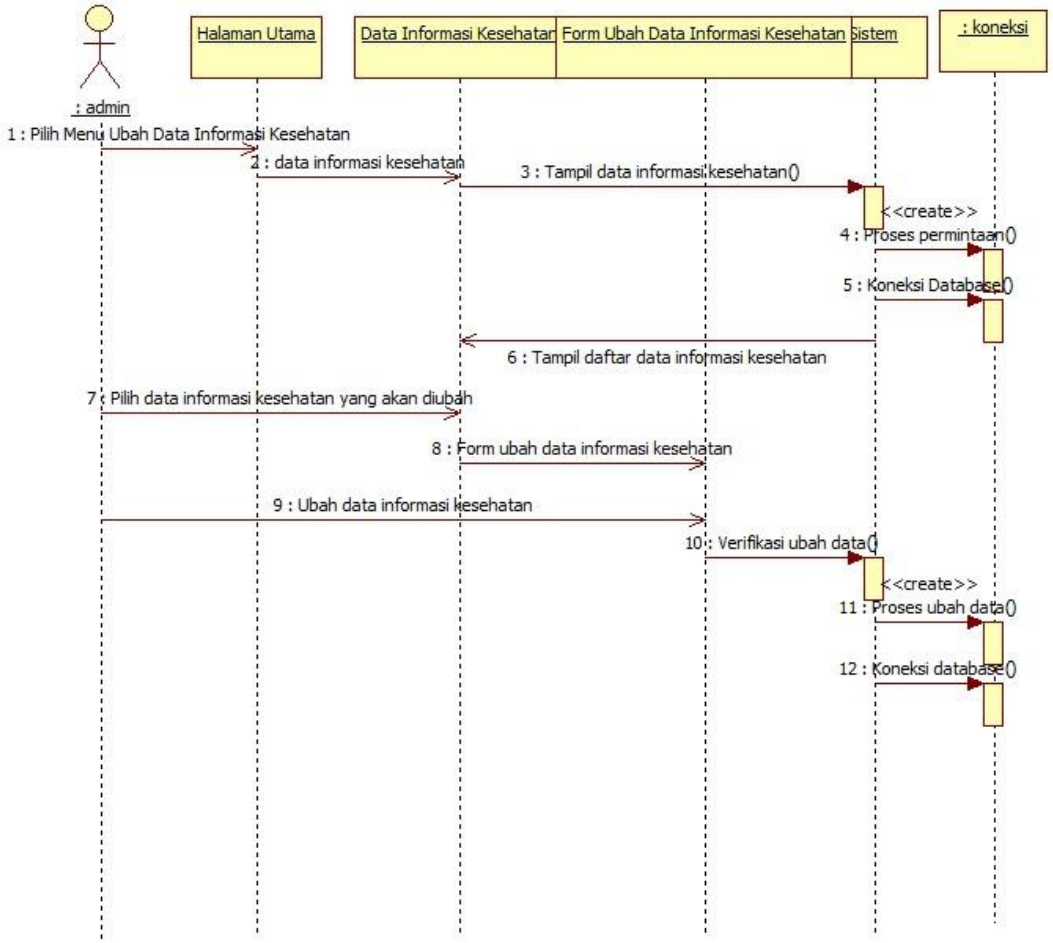
Gambar 3.36 merupakan sequence diagram input data informasi kesehatan. Proses ini melibatkan admin, aplikasi, serta dokumen html/text yang akan diinputkan kedalam aplikasi untuk diolah lebih lanjut.



Gambar 3.36 Sequence Diagram *Input* Data Informasi Kesehatan

3.3.6.5 Sequence Diagram *Edit* Data Informasi Kesehatan

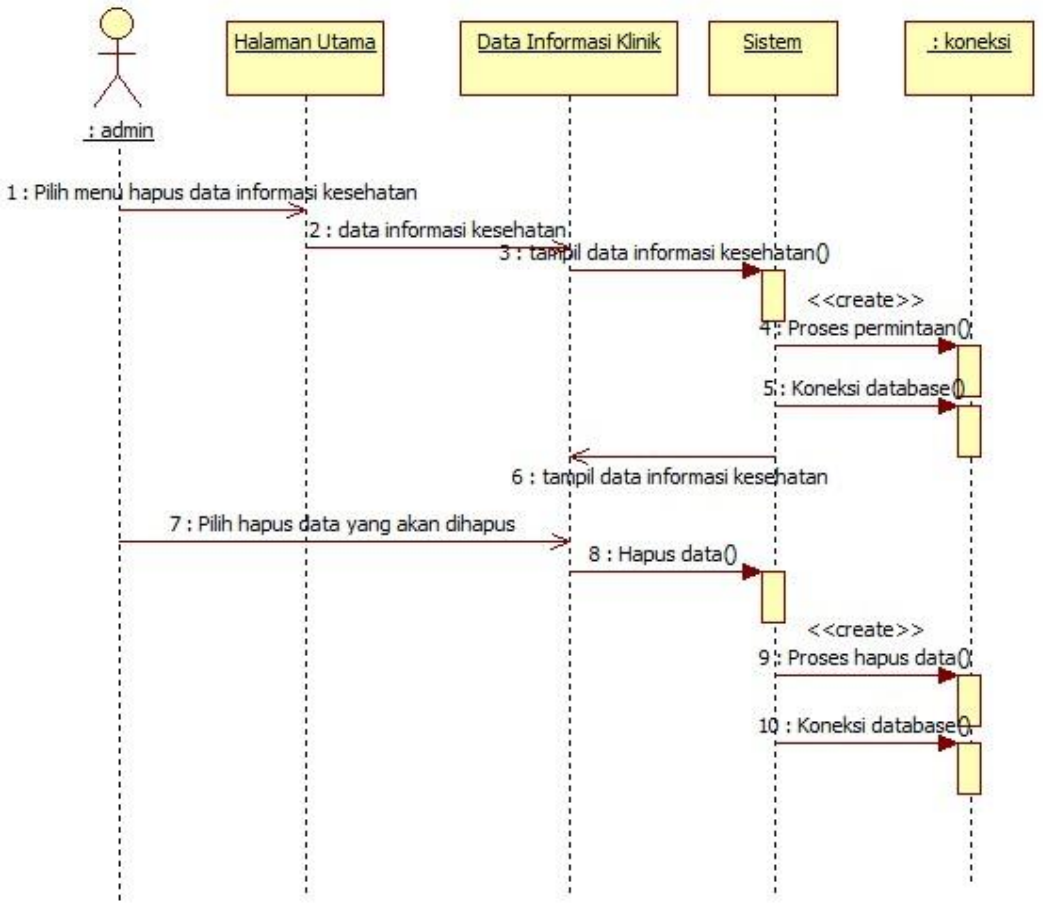
Gambar 3.37 merupakan proses pembaharuan (*edit*) dokumen html/text yang berisi tentang data informasi klinik, yang mana dalam prosesnya melibatkan admin, aplikasi dan dokumen html/text.



Gambar 3.37 Sequence Diagram *Edit* Data Informasi Kesehatan

3.3.6.6 Sequence Diagram Hapus Data Informasi Kesehatan

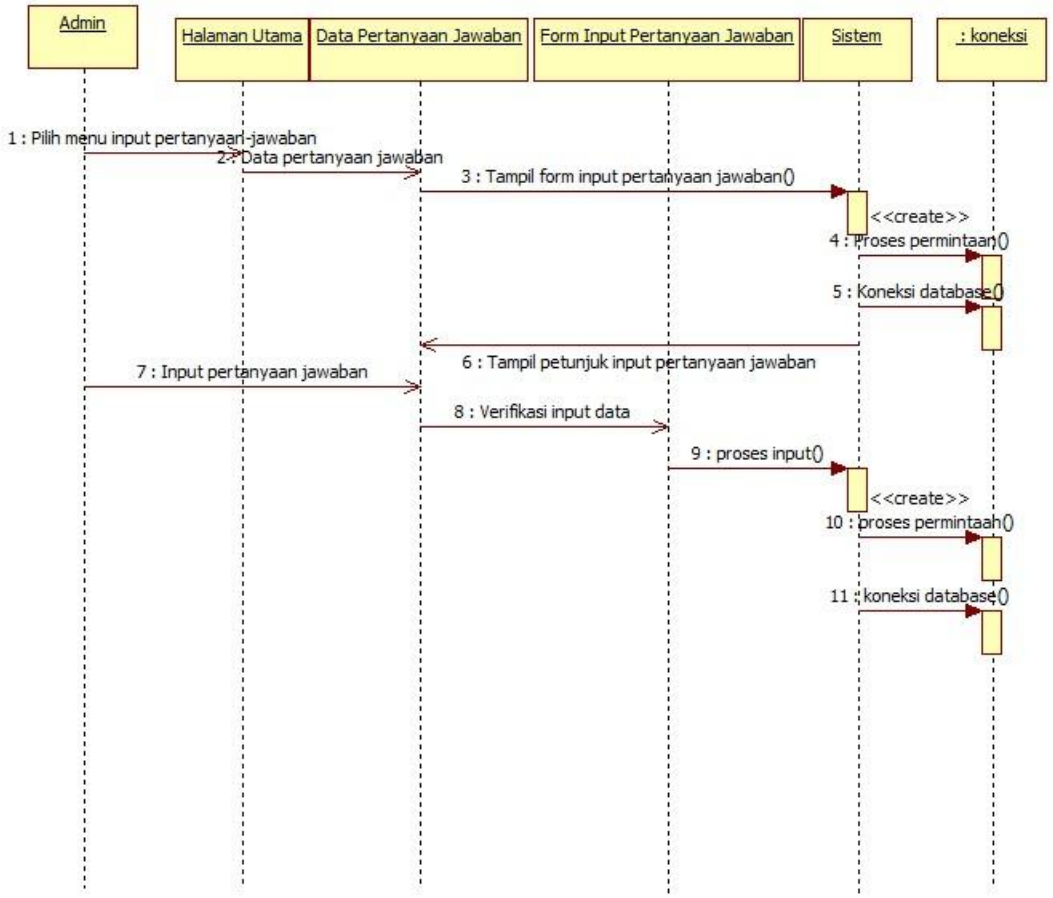
Gambar 3.38 merupakan proses penghapusan data informasi klinik, yang mana dalam prosesnya melibatkan admin, aplikasi, dan dokumen html/text.



Gambar 3.38 Sequence Diagram Hapus Data Informasi Kesehatan

3.3.6.7 Sequence Diagram Input Trigger – Reply Informasi Kesehatan Pada *chatbot* dokter *virtual*

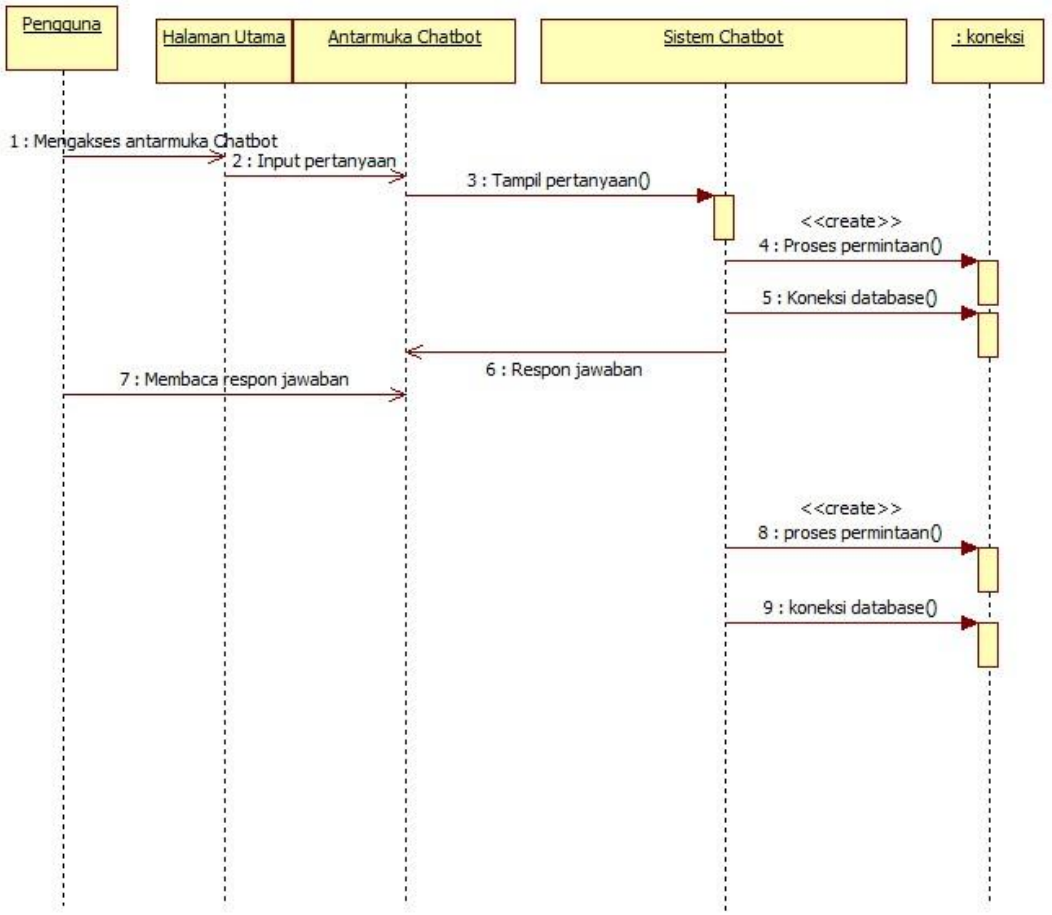
Gambar 3.39 merupakan sequence diagram input data informasi kesehatan. Proses ini melibatkan admin, aplikasi, serta dokumen html/text yang akan diinputkan kedalam aplikasi untuk diolah lebih lanjut.



Gambar 3.39 Sequence Diagram Input Pertanyaan Jawaban Trigger – Reply Informasi Kesehatan Pada *chatbot* dokter *virtual*

3.3.6.8 Sequence Diagram Melihat dan Mencari Informasi Kesehatan Melalui *Chatbot* Dokter *Virtual*

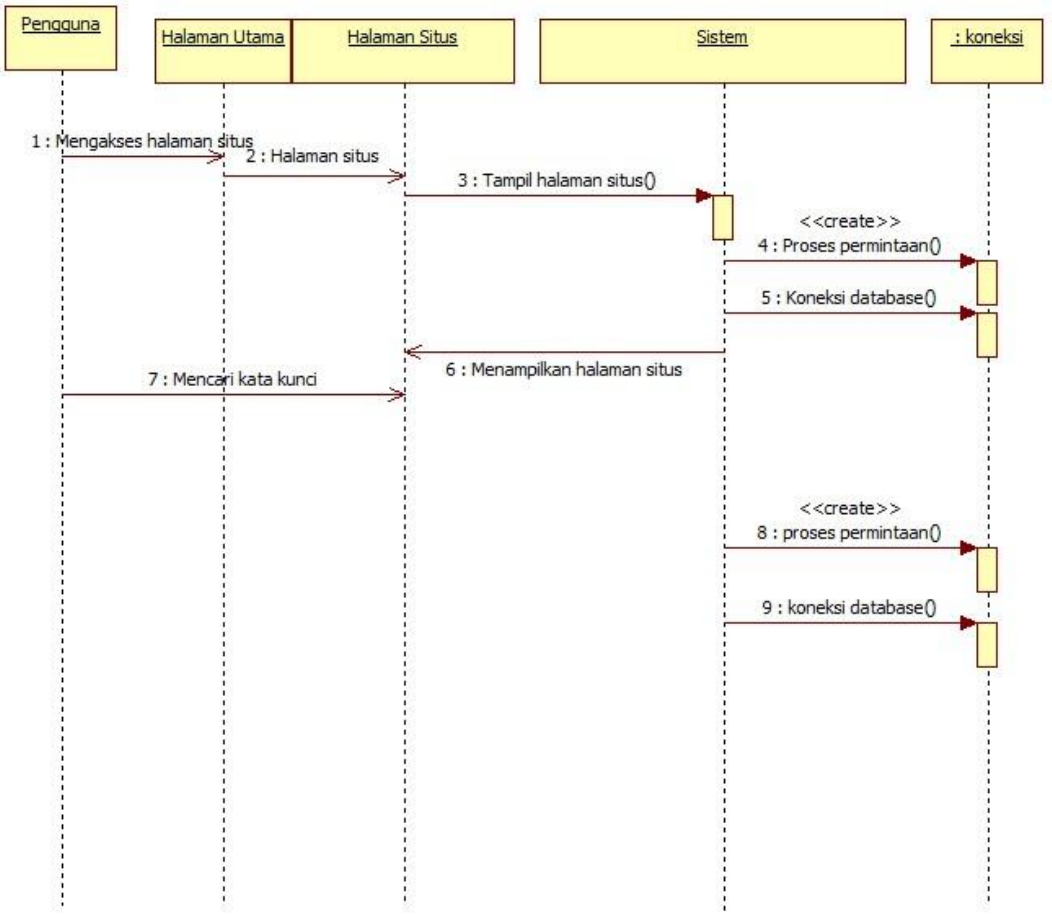
Gambar 3.40 adalah proses untuk melihat dan mencari informasi kesehatan melalui *chatbot*, dimana prosesnya melibatkan pengguna dan sistem *chatbot* dokter *virtual*



Gambar 3.40 Sequence Diagram Melihat dan Mencari Informasi Kesehatan Melalui *Chatbot* Dokter V*irtual*

3.3.6.9 Sequence Diagram Melihat dan Mencari Informasi Kesehatan Melalui Halaman Situs

Gambar 3.41 adalah proses untuk melihat dan mencari informasi kesehatan melalui halaman situs, dimana prosesnya melibatkan pengguna dan sistem chatbot.



Gambar 3.41 Sequence Diagram Melihat dan Mencari Informasi Kesehatan Melalui Halaman Situs

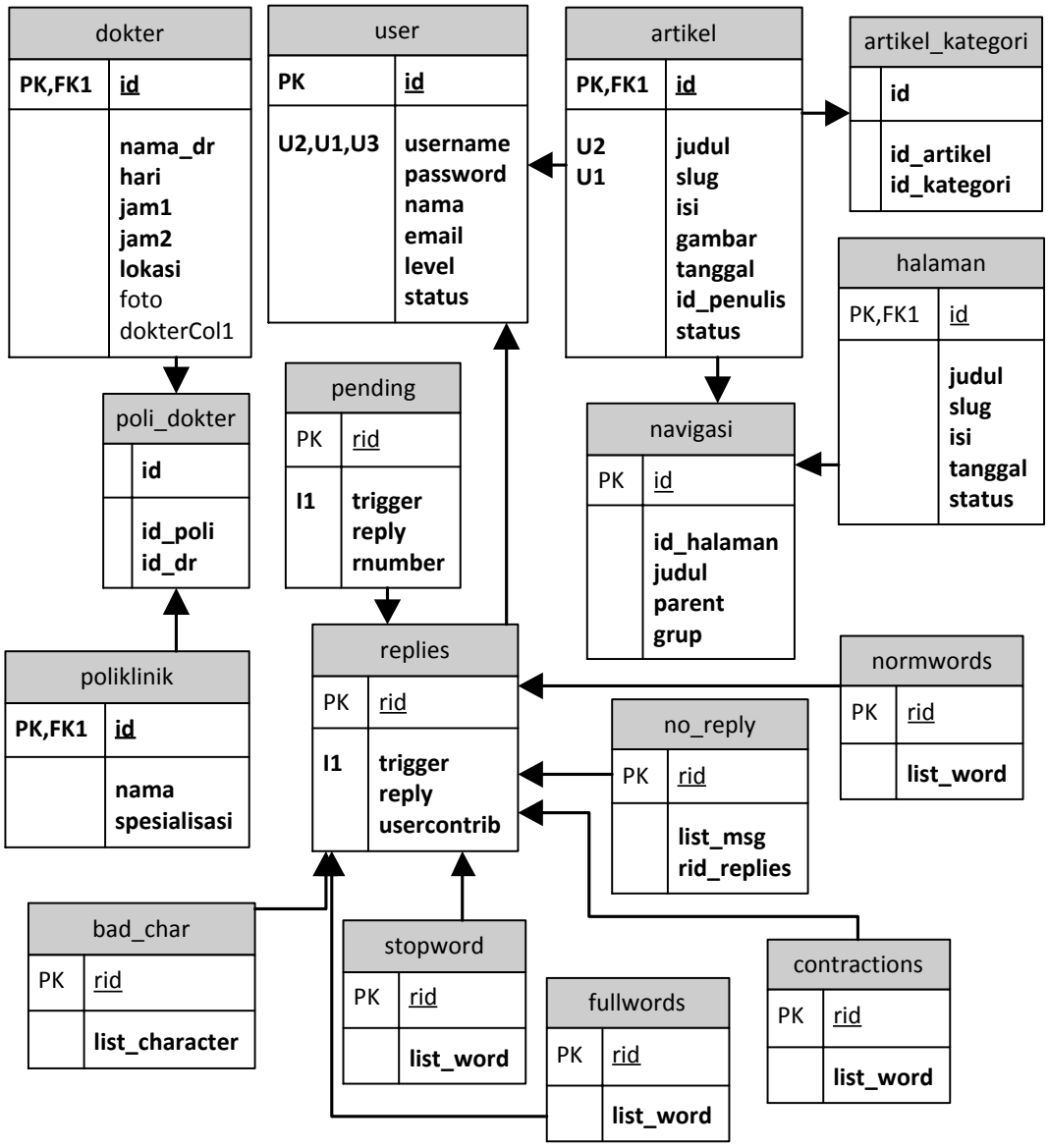
3.3.7 Perancangan Basis Data

Perancangan basis data yaitu menciptakan atau merancang data yang terhubung dan disimpan secara bersama-sama. Untuk menggambarkannya digunakanlah diagram relasi dan dan struktur table. Dari dua hasil tersebut, implementasikan basis data akan bisa dikerjakan.

3.3.7.1 Diagram Relasi

Model data relasional merupakan model data dimana hubungan antar data, arti data dan batasannya dijelaskan dengan baris dan kolom. Secara formal, ke semuanya itu digambarkan ke dalam diagram relasi dan diagram skema.

Adapun untuk lebihnya tentang diagram relasi pada sistem *chatbot* dokter *virtual* gambar 3.42 sebagai berikut



Gambar 3.42 Diagram Relasi Aplikasi *Chatbot* Dokter *Virtual*

3.3.7.2 Struktur Tabel

Dibawah ini adalah struktur tabel yang digunakan dalam aplikasi web klinik yang terbagi ke dalam beberapa tabel, sebagai berikut :

1. Tabel User

a. Nama Tabel : user
b. Primary Key : id
c. Jumlah Field : 6
d. Keterangan :Tabel berisi data user admin

Tabel 3.20 Struktur Tabel *User*

Nama Field	Type	Length	Keterangan
Id	Int	11	*Primary key*
username	Varchar	100	
password	varchar	255	
Nama	varchar	100	
Level	enum		“admin”, “user”
Status	Enum		“no”, “yes”

2. Tabel Artikel

Tabel content web

a. Nama Tabel : artikel
b. Primary Key : id
c. Jumlah Field : 8
d. Keterangan : Tabel berisi data artikel informasi kesehatan

Tabel 3.21 Struktur Tabel Artikel

Nama Field	Type	Length	Keterangan
Id	Int	11	*primary key*
Judul	Varchar	100	
Slug	Varchar	250	
Isi	Text	50	
gambar	Varchar	250	
tanggal	Datetime		
id_penulis	Int		
Status	Enum		“draft”, “live”

3. Tabel Kategori Artikel

a. Nama Tabel : artikel_kategori
b. Primary Key : id
c. Jumlah Field : 3
d. Keterangan : Tabel berisi data kategori artikel informasi kesehatan

Tabel 3.22 Struktur Tabel Kategori Artikel

Nama Field	Type	Length	Keterangan
Id	Int	11	*primary key*
id_artikel	Int	11	*foreign key*
id_kategori	Int	11	*foreign key*

4. Tabel Dokter

a. Nama Tabel : dokter
b. Primary Key : id
c. Jumlah field : 7
d. Keterangan : Tabel berisi data dokter

Tabel 3.23 Struktur Tabel Dokter

Nama Field	Type	Length	Keterangan
Id	Int	11	*primary key*
nama_dr	Varchar	100	
Hari	Varchar	50	
jam1	Time		
jam2	Time		
Lokasi	Varchar	100	
Foto	Varchar	255	

5. Tabel Dokter Poliklinik

a. Nama File : poli_dokter
b. Primary Key : id
c. Jumlah Field : 3
d. Keterangan : Tabel berisi data relasi poliklinik dan dokter

Tabel 3.24 Struktur Tabel Dokter Poliklinik

Nama Feld	Type	Length	Keterangan
Id	Int	11	*primary key*
id_poli	Int	11	*foreign key*
id_dokter	Int	11	*foreign key*

6. Tabel Poliklinik

a. Nama File : poliklinik

b. Primary Key : id

c. Jumlah Field : 3

d. Keterangan : Tabel berisi data poliklinik

Tabel 3.25 Struktur Tabel Poliklinik

Nama Field	Type	Length	Keterangan
Id	int	11	*primary key*
Nama	varchar	100	
spesialisasi	varchar	100	

7. Tabel Halaman

a. Nama File : halaman

b. Primary Key : id

c. Jumlah Field : 6

d. Keterangan : Tabel berisi data halaman web

Tabel 3.26 Struktur Tabel Halaman

Nama Field	Type	Length	Keterangan
Id	int	11	*primary key*
Judul	varchar	100	
Slug	varchar	250	
Isi	text	50	
tanggal	datetime		
Status	enum		“draft”, “live”

8. Tabel Navigasi

a. Nama File : halaman

b. Primary Key : id

c. Jumlah Field : 6

d. Keterangan : Tabel berisi data navigasi halaman web

Tabel 3.27 Struktur Tabel Navigasi

Nama Field	Type	Length	Keterangan
Id	int	11	*primary key*
id_halaman	int	11	*foreign key*
Judul	varchar	250	
Parent	int	11	
Grup	enum		“header”, “side”

9. Tabel Pending

a. Nama File : pending

b. Primary Key : rid

c. Jumlah Field : 4

d. Keterangan : Tabel berisi data trigger reply chatbot (learning)

Tabel 3.28 Struktur Tabel Pending

Nama Field	Type	Length	Keterangan
Trigger	text		*primary key*
Reply	text		
rnumber	int	4	
Rid	int	10	

9. Tabel Replies

a. Nama File : replies

b. Primary Key : rid

c. Jumlah Field : 4

d. Keterangan : Tabel berisi data trigger reply chatbot

Tabel 3.29 Struktur Tabel Replies

Nama Field	Type	Length	Keterangan
Trigger	text		*primary key*
Reply	text		
usercontrib	int	4	
Rid	int	10	

10. Tabel No_Reply

a. Nama File : no_reply

b. Primary Key : rid

c. Jumlah Field : 3

d. Keterangan : Tabel berisi daftar respon chatbot jika tidak tersedia reply

Tabel 3.30 Struktur Tabel No_Reply

Nama Field	Type	Length	Keterangan
Rid	int		*primary key*
list_msg	text	30	
rid_replies	int	4	

11. Tabel Bad_Char

a. Nama File : bad_char
b. Primary Key : rid
c. Jumlah Field : 2
d. Keterangan : Tabel berisi daftar karakter yang tidak digunakan chatbot

Tabel 3.31 Struktur Tabel Bad_Char

Nama Field	Type	Length	Keterangan
Rid	int		*primary key*
list_character	text	1	

12. Tabel Stopword

a. Nama File : stopword
b. Primary Key : rid
c. Jumlah Field : 2
d. Keterangan : Tabel berisi daftar kata yang tidak digunakan chatbot

Tabel 3.32 Struktur Tabel Stopword

Nama Field	Type	Length	Keterangan
Rid	Int		*primary key*
list_word	Text	20	

14. Tabel Fullwords

a. Nama File : fullwords
b. Primary Key : rid
c. Jumlah Field : 2
d. Keterangan : Tabel berisi daftar kata sambung yang digunakan dalam normalisasi kata

Tabel 3.33 Struktur Tabel Fullwords

Nama Field	Type	Length	Keterangan
Rid	Int		*primary key*
list_word	Text	20	

15. Tabel Contractions

a. Nama File : constractions
b. Primary Key : rid
c. Jumlah Field : 2
d. Keterangan : Tabel berisi daftar imbuhan yang digunakan dalam normalisasi kata

Tabel 3.34 Struktur Tabel Constractions

Nama Field	Type	Length	Keterangan
rid	Int		*primary key*
list_word	Text	20	

16. Tabel Normwords

e. Nama File : normwords

f. Primary Key : rid

g. Jumlah Field : 2

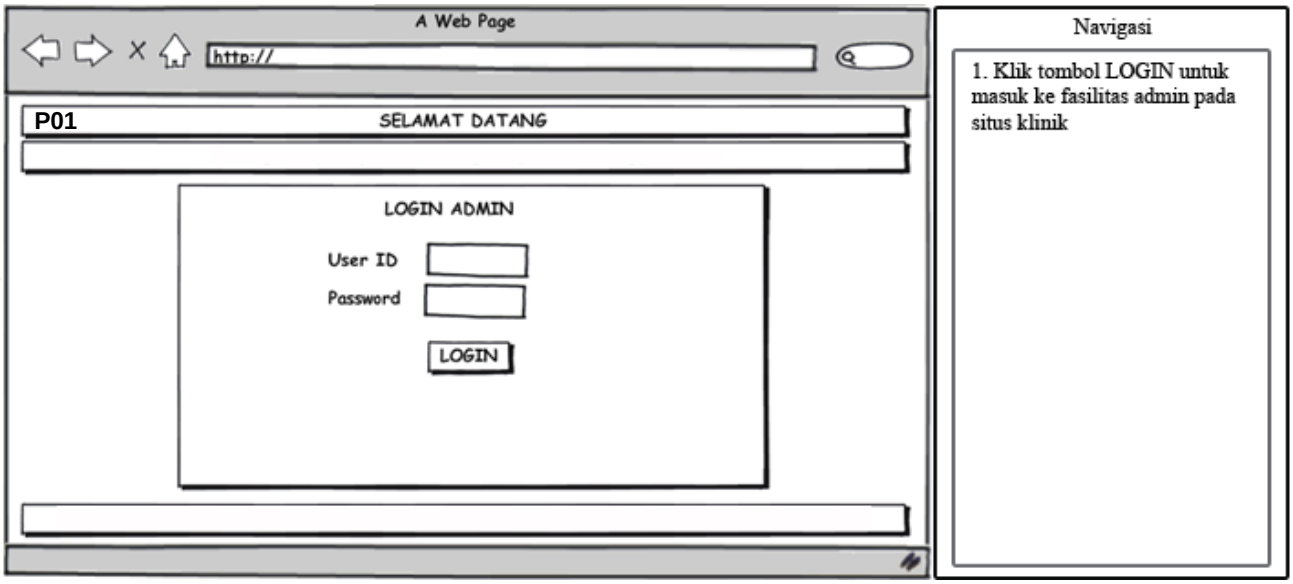
h. Keterangan : Tabel berisi daftar hasil normalisasi kata

Tabel 3.35 Struktur Tabel Normwords

Nama Field	Type	Length	Keterangan
Rid	Int		*primary key*
list_word	Text	20	

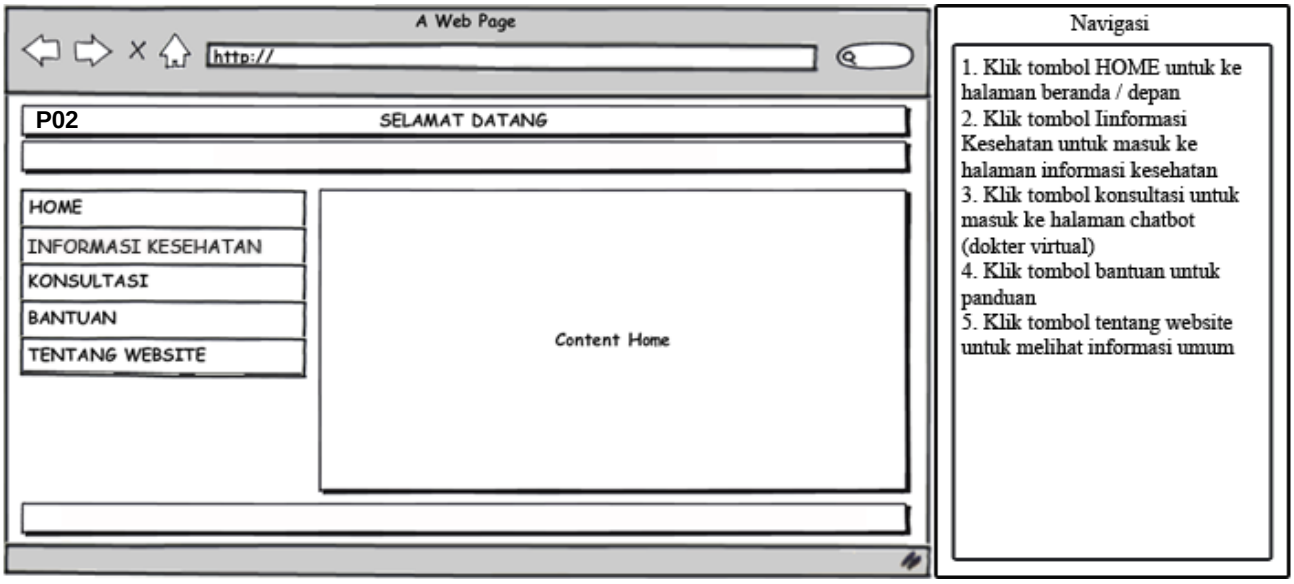
3.3.8 Perancangan Antar Muka Aplikasi *Chatbot* Dokter *Virtual*

1. Nama Dialog Screen : Halaman Login
   Fungsi : Halaman awal untuk mengakses halaman admin
   Nomor : P01



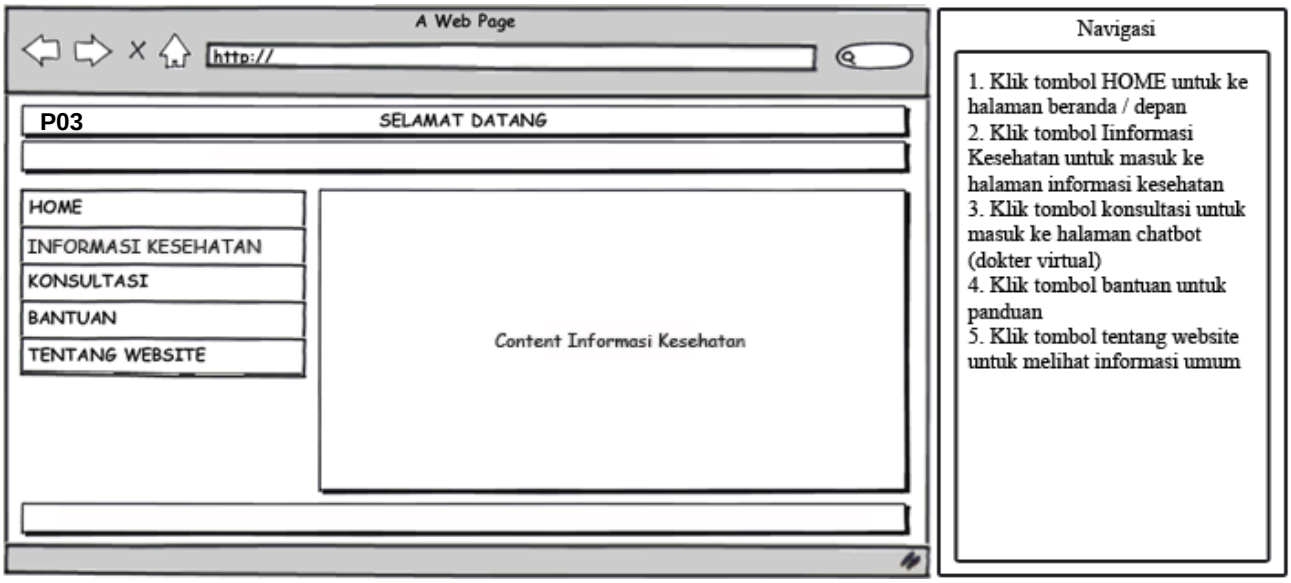
Gambar 3.43 Rancangan antarmuka halaman login

2. Nama Dialog Screen : Halaman awal *user*
   Fungsi : Halaman awal untuk *user*
   Nomor : P02



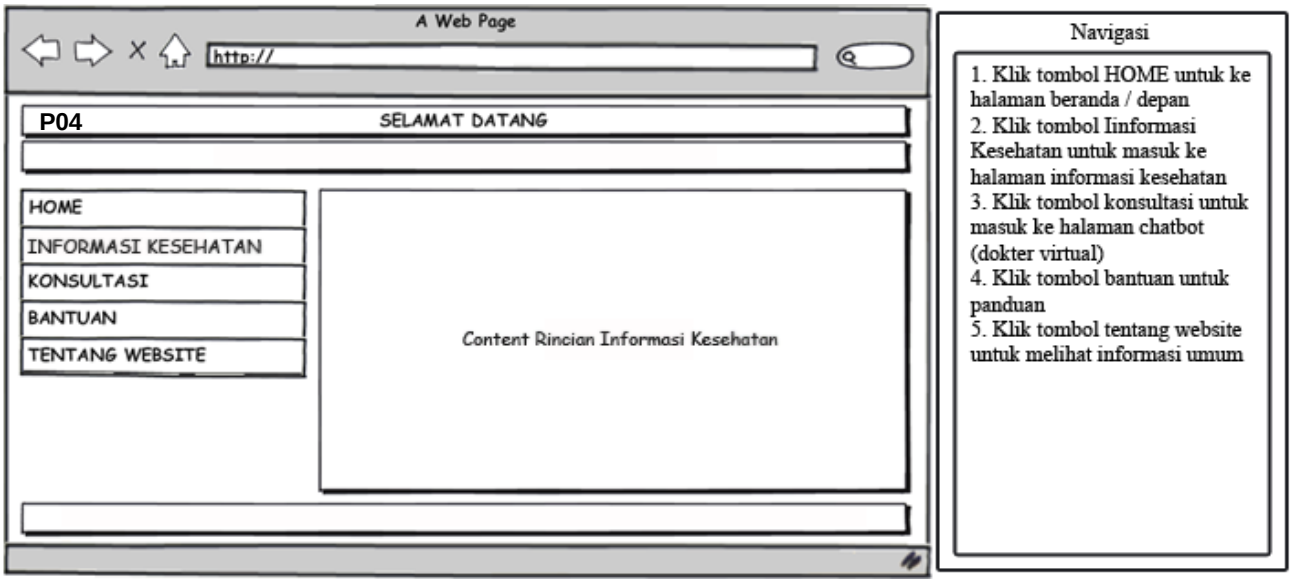
Gambar 3.44 Rancangan Antarmuka Halaman Beranda

3. Nama Dialog Screen : Halaman informasi kesehatan
   Fungsi : Halaman untuk melihat dan mencari informasi kesehatan untuk pengguna
   Nomor : P03



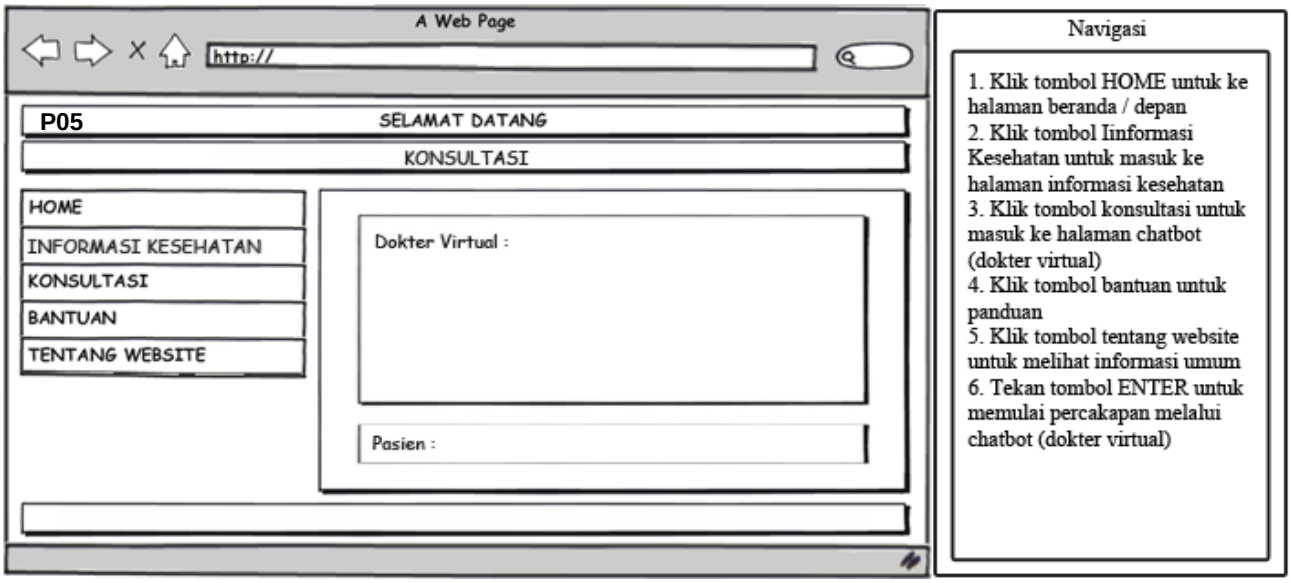
Gambar 3.45 Rancangan Antarmuka Halaman Informasi Kesehatan

4. Nama Dialog Screen : Halaman rincian informasi kesehatan
   Fungsi : Halaman untuk melihat dan mencari rincian informasi kesehatan untuk pengguna
   Nomor : P04



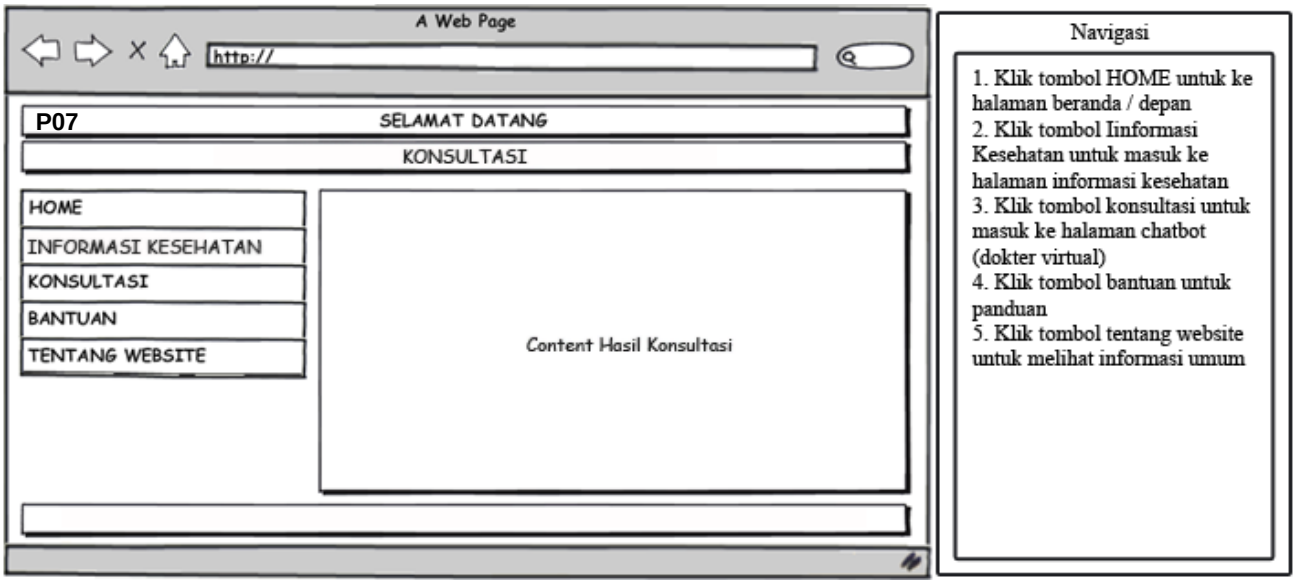
Gambar 3.46 Rancangan antarmuka halaman rincian informasi kesehatan

5. Nama Dialog Screen : Halaman konsultasi informasi kesehatan
   Fungsi : Halaman untuk melakukan konsultasi informasi kesehatan dengan *Chatbot* Dokter *Virtual*
   Nomor : P05



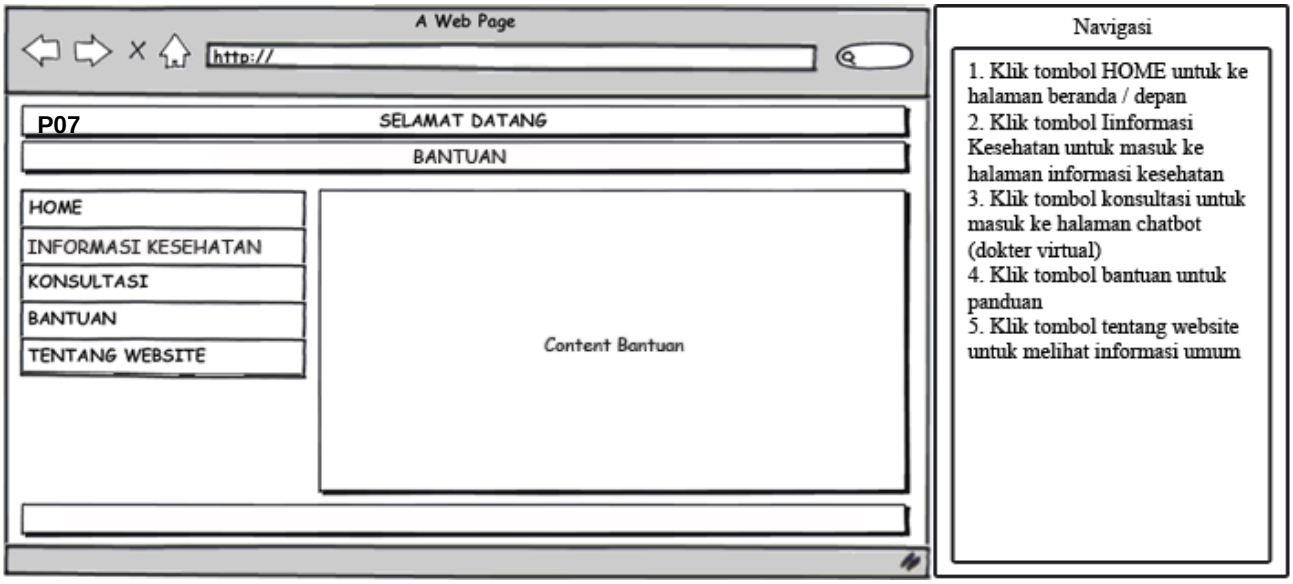
Gambar 3.47 Rancangan Antarmuka Halaman Konsultasi *Chatbot* Dokter *Virtual*

6. Nama Dialog Screen : Halaman hasil konsultasi informasi kesehatan
   Fungsi : Halaman untuk menampilkan hasil konsultasi informasi kesehatan dengan Chatbot
   Nomor : P06



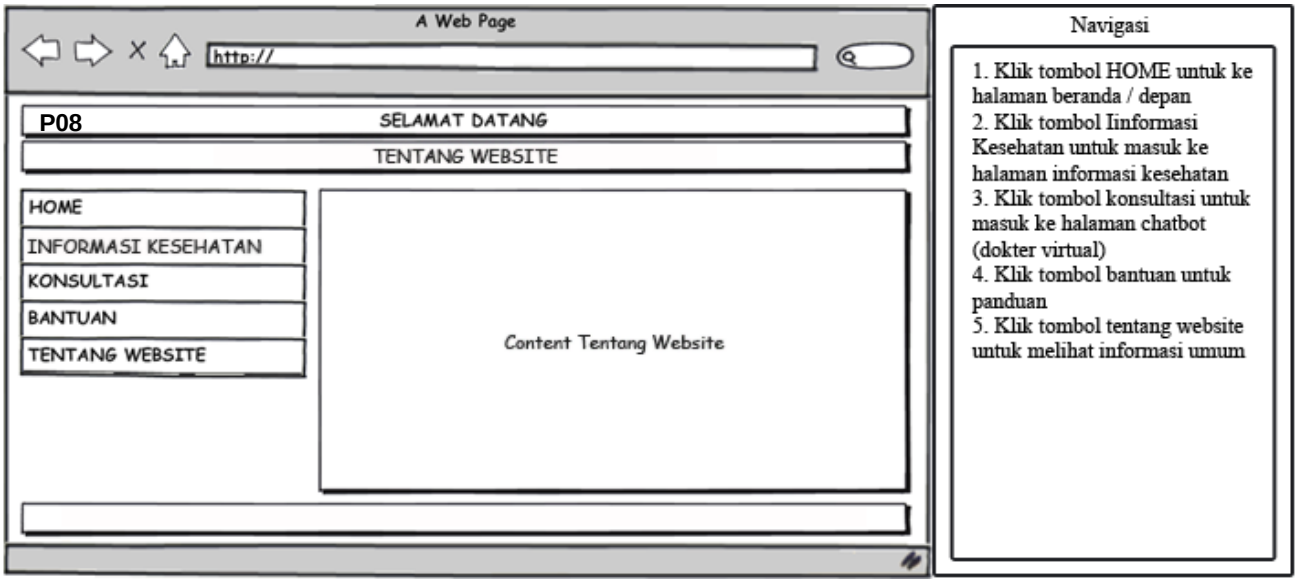
Gambar 3.48 Rancangan Antarmuka Halaman Hasil Konsultasi Informasi Kesehatan dengan *Chatbot* Dokter *Virtual*

7. Nama Dialog Screen : Halaman bantuan
   Fungsi : Halaman untuk melihat bantuan penggunaan website
   Nomor : P07



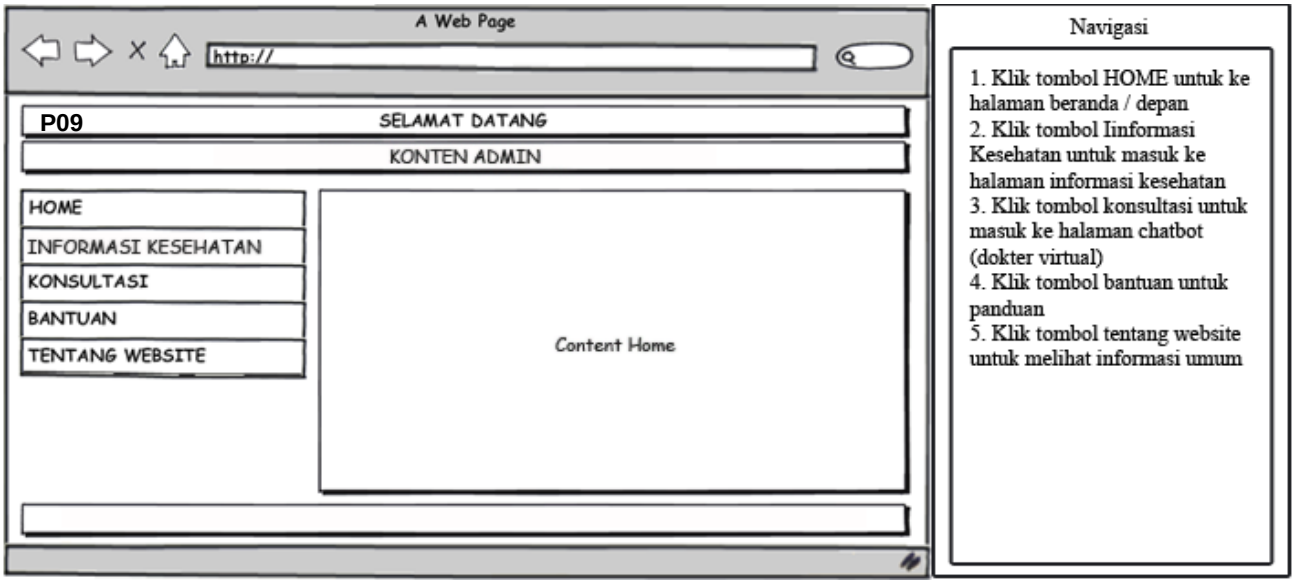
Gambar 3.49 Rancangan Antarmuka Halaman Bantuan

8. Nama Dialog Screen : Halaman tentang website
   Fungsi : Halaman untuk melihat gambaran umum tentang website
   Nomor : P08



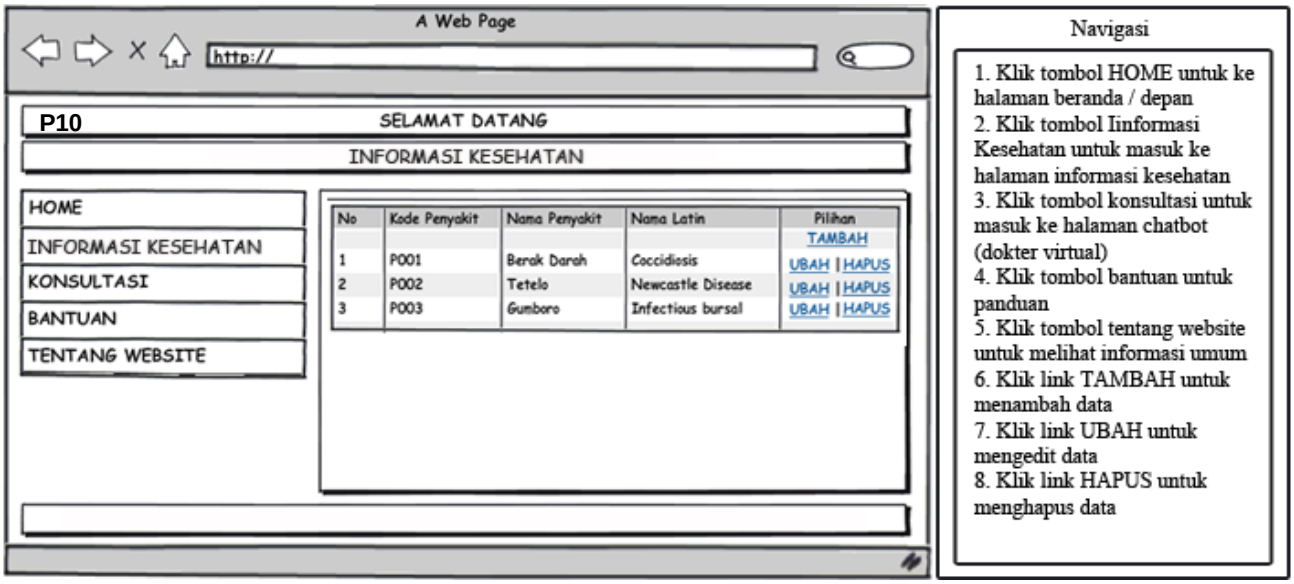
Gambar 3.50 Rancangan Antarmuka Halaman Tentang Website

9. Nama Dialog Screen : Halaman awal admin
   Fungsi : Halaman awal untuk admin
   Nomor : P09



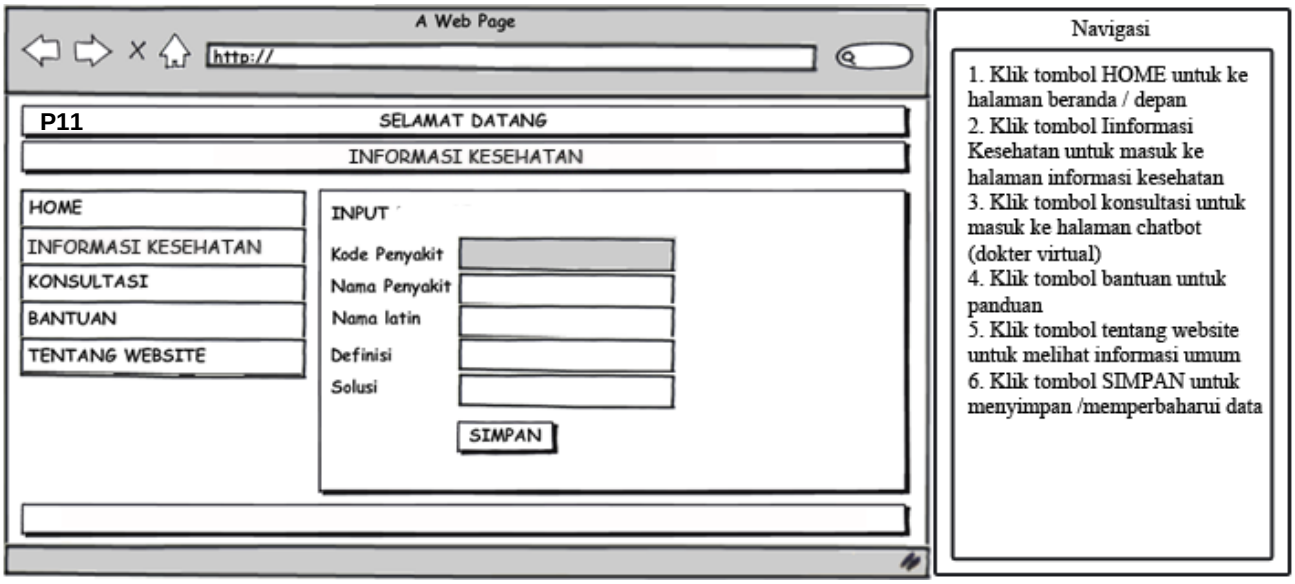
Gambar 3.51 Rancangan Antarmuka Halaman Awal Admin

10. Nama Dialog Screen : Halaman daftar informasi kesehatan
    Fungsi : Halaman untuk melihat daftar informasi kesehatan
    Nomor : P10



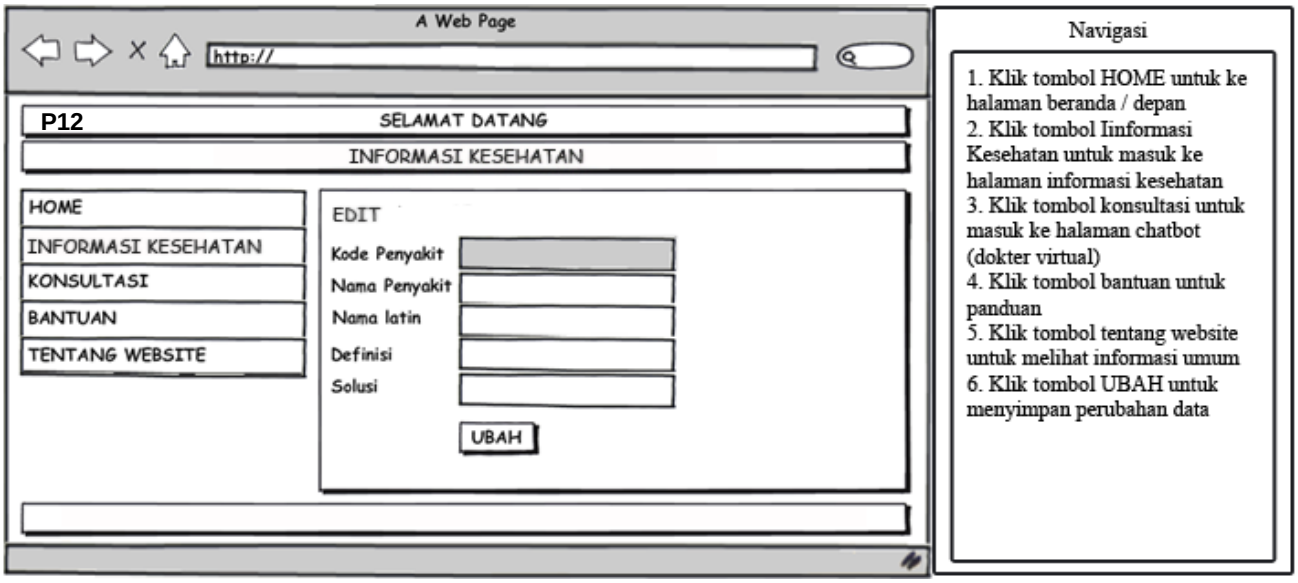
Gambar 3.52 Rancangan Antarmuka Halaman Daftar Data Informasi Kesehatan

11. Nama Dialog Screen : Halaman tambah data informasi kesehatan
    Fungsi : Halaman untuk menambahkan data informasi kesehatan
    Nomor : P11



Gambar 3.53 Rancangan Antarmuka Halaman Input Data Informasi Kesehatan

12. Nama Dialog Screen : Halaman ubah data informasi kesehatan
    Fungsi : Halaman untuk mengubah data informasi kesehatan
    Nomor : P12

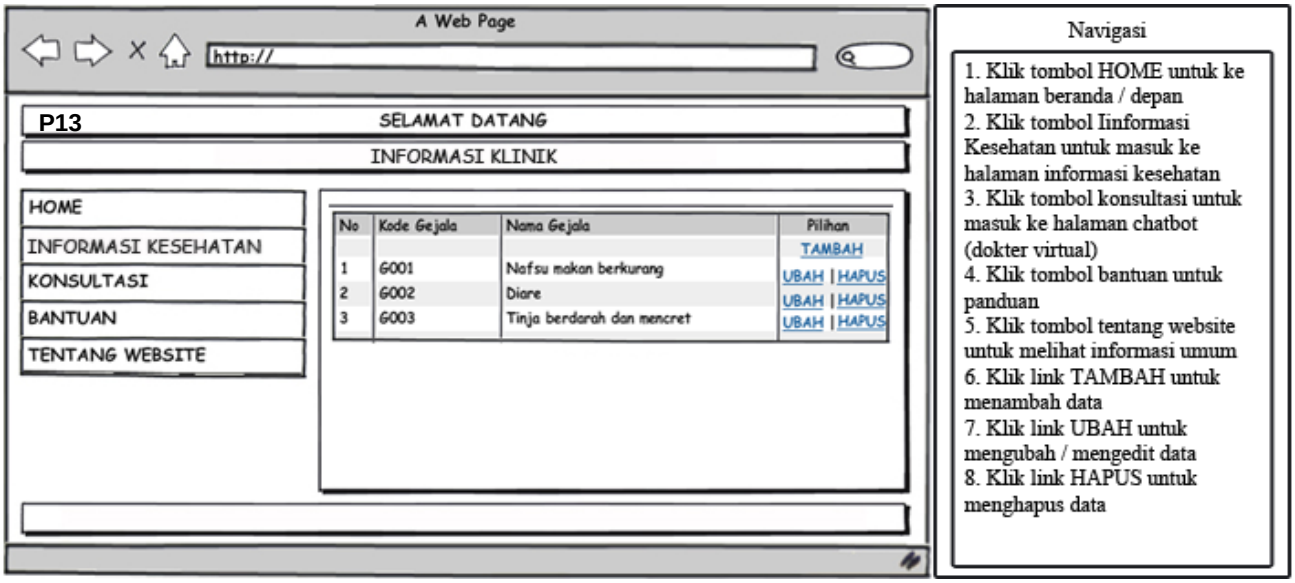


Gambar 3.54 Rancangan Antarmuka Halaman *Edit* Data Informasi Kesehatan

13. Nama Dialog Screen : Halaman daftar informasi klinik

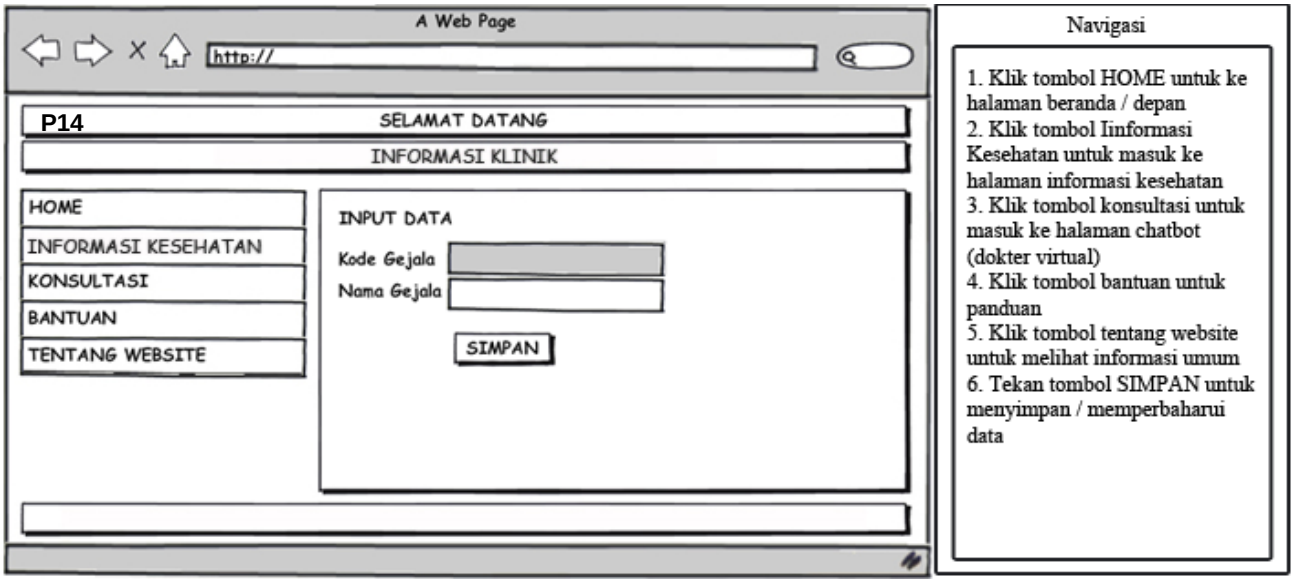
    Fungsi : Halaman untuk melihat daftar informasi klinik

    Nomor : P13



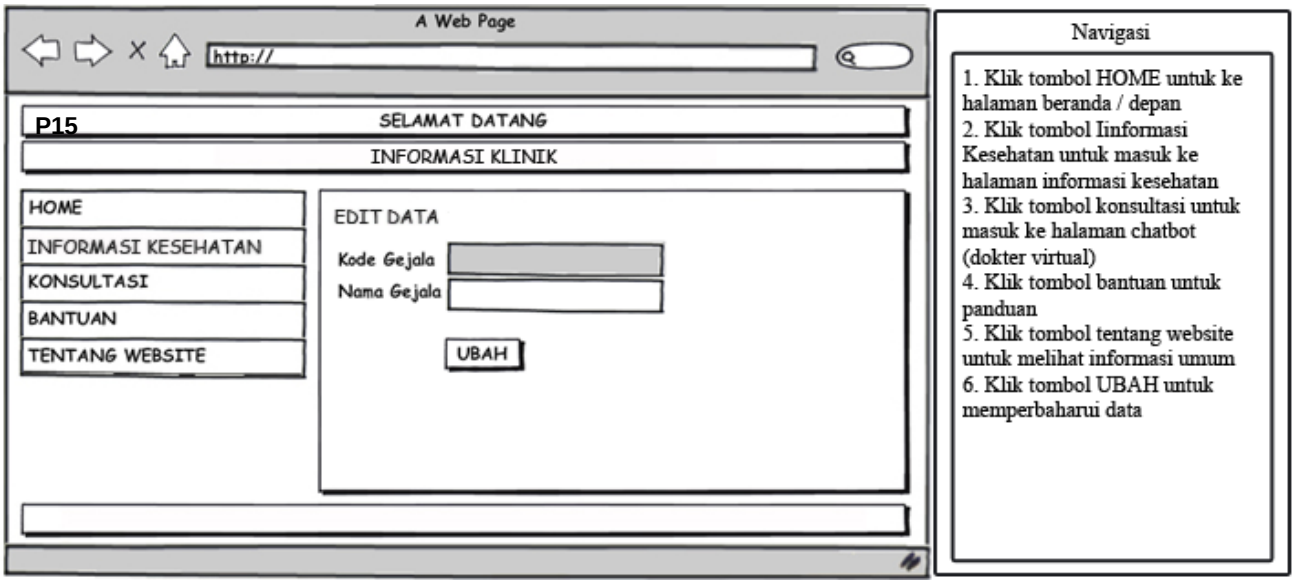
Gambar 3.55 Rancangan antarmuka halaman daftar informasi klinik

14. Nama Dialog Screen : Halaman tambah data informasi klinik
    Fungsi : Halaman untuk menambah data informasi klinik
    Nomor : P14



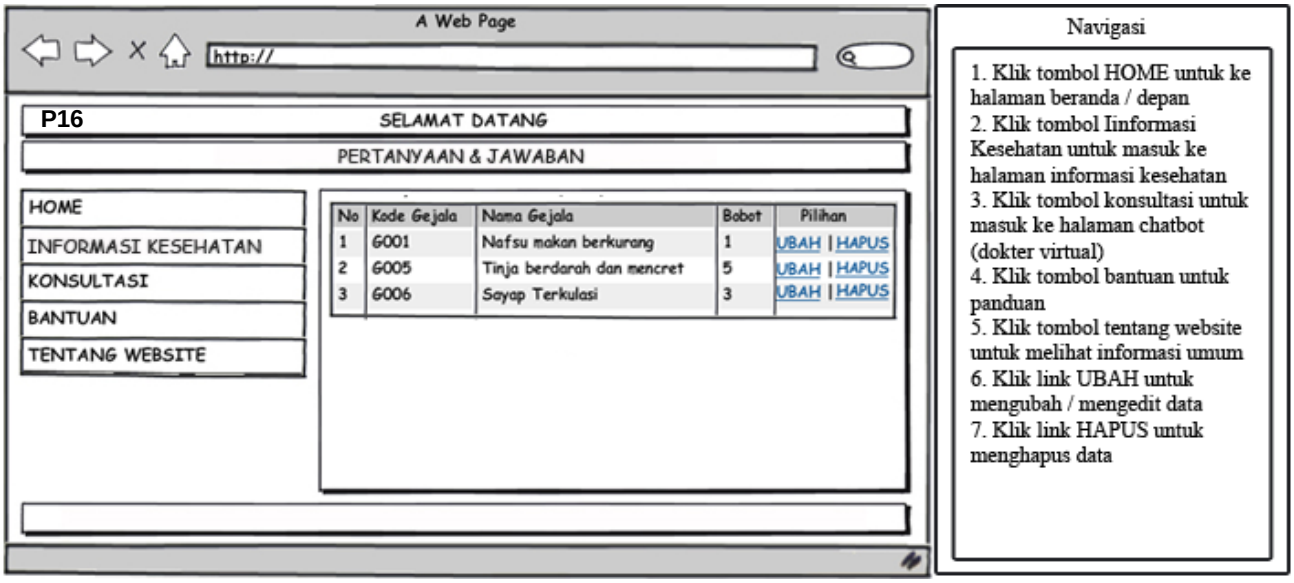
Gambar 3.56 Rancangan Antarmuka Halaman *Input* Data Informasi Klinik

15. Nama Dialog Screen : Halaman ubah data informasi klinik
    Fungsi : Halaman untuk mengubah data informasi klinik
    Nomor : P15



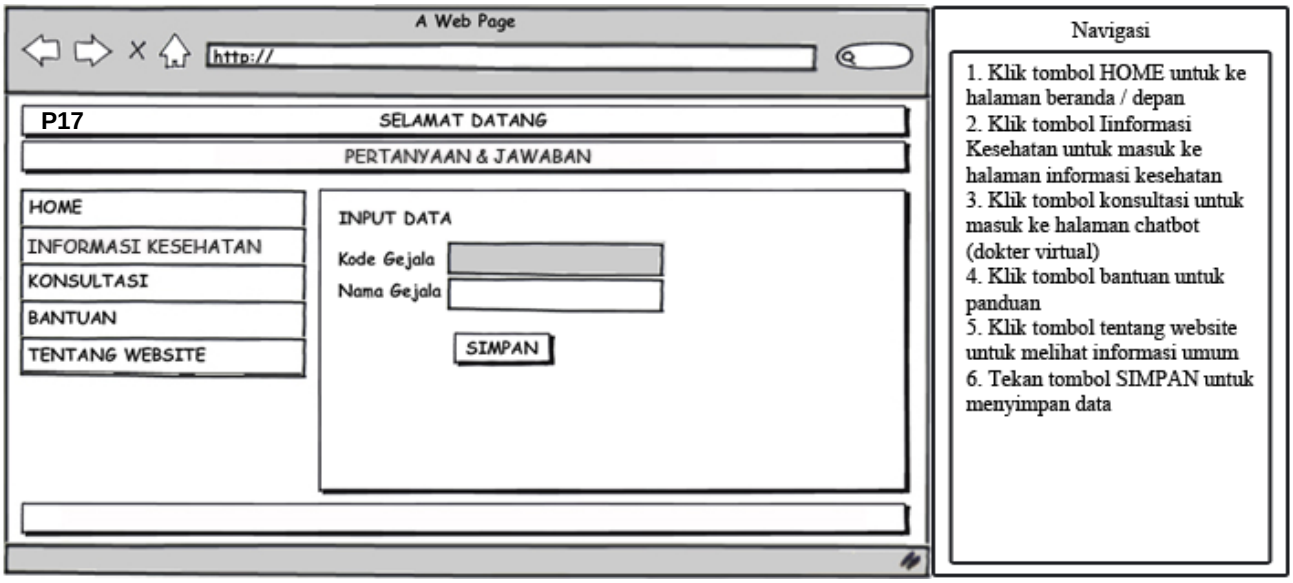
Gambar 3.57 Rancangan Antarmuka Halaman *Edit* Data Informasi Klinik

16. Nama Dialog Screen : Halaman daftar pertanyaan jawaban
    Fungsi : Halaman untuk daftar pertanyaan jawaban
    Nomor : P16



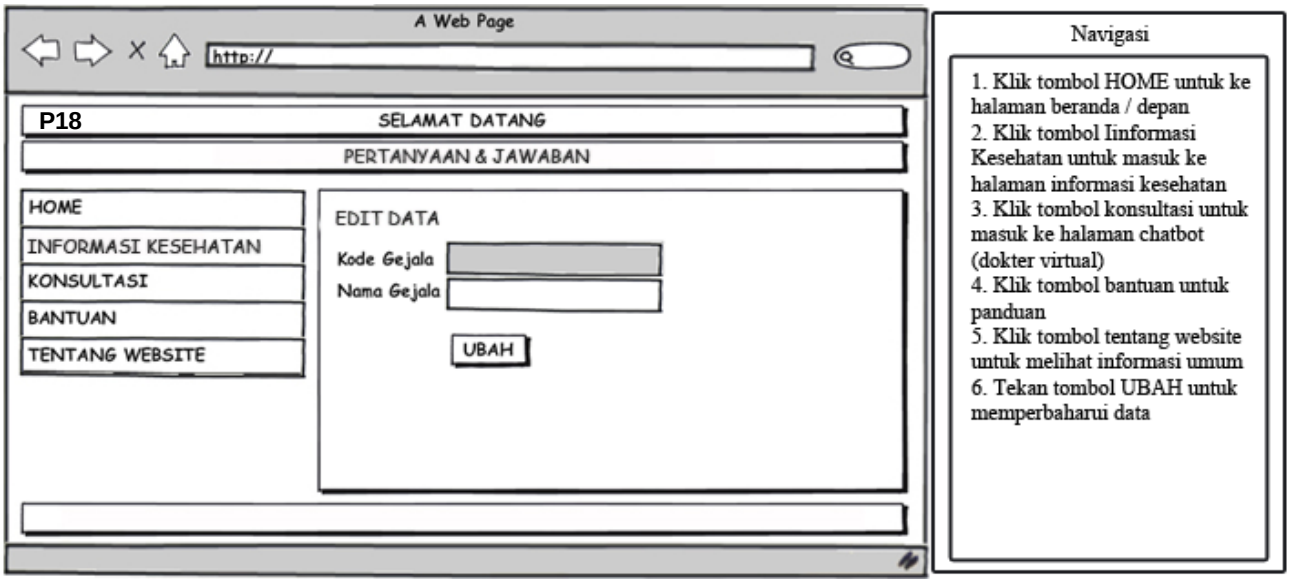
Gambar 3.58 Rancangan Antarmuka Halaman Daftar Pertanyaan Jawaban

17. Nama Dialog Screen : Halaman tambah pertanyaan jawaban chatbot
    Fungsi : Halaman untuk menambah pertanyaan jawaban *Chatbot* Dokter *Virtual*
    Nomor : P17



Gambar 3.59 Rancangan antarmuka halaman tambah pertanyaan jawaban *Chatbot* Dokter *Virtual*

18. Nama Dialog Screen : Halaman ubah pertanyaan dan jawaban
    Fungsi : Halaman untuk mengubah pertanyaan dan jawaban
    Bentuk : P18

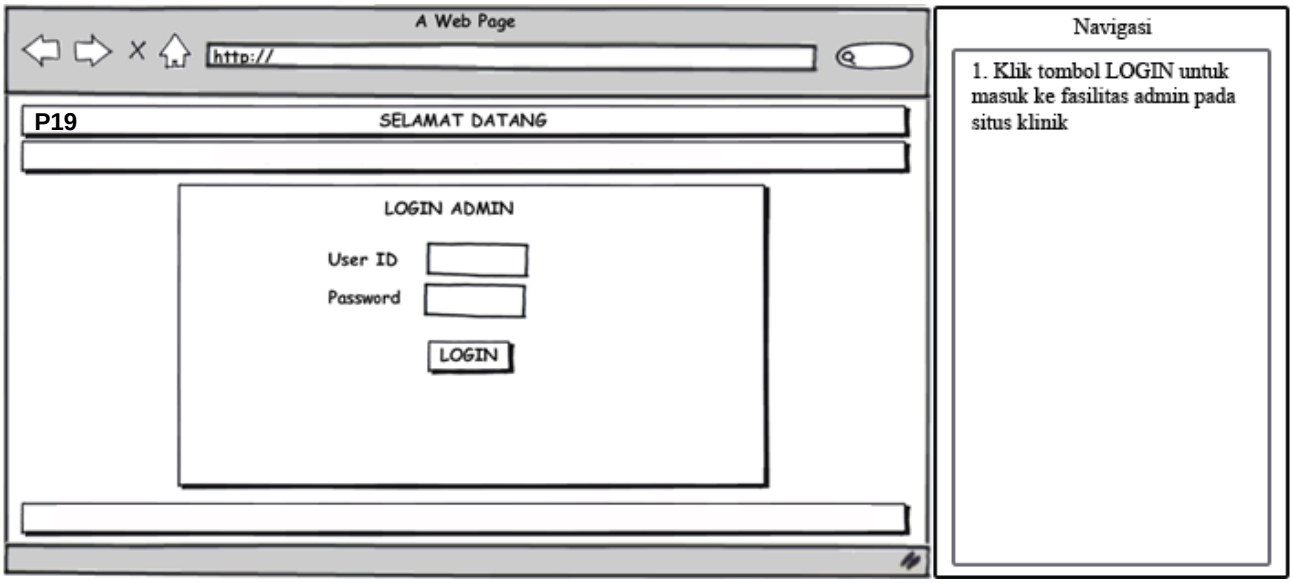


Gambar 3.60 Rancangan Antarmuka Halaman Ubah Data Pertanyaan Jawaban

19. Nama Dialog Screen : Halaman *Login* Admin

Fungsi : Halaman awal untuk mengakses halaman admin

Nomor : P19

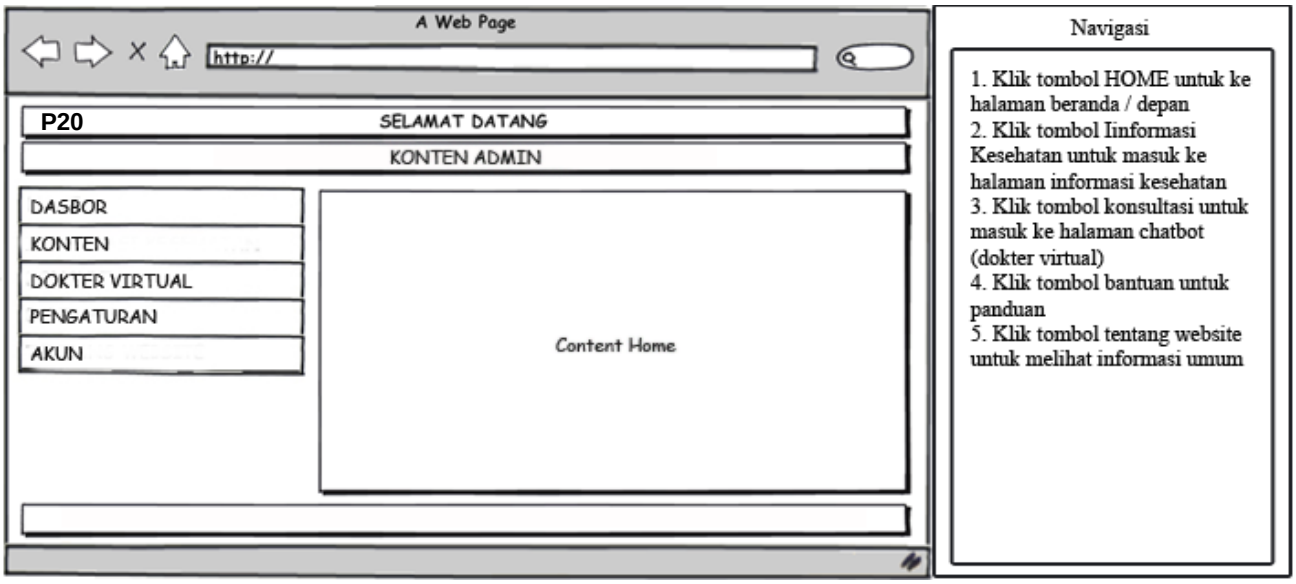


Gambar 3.61 Rancangan antarmuka halaman *login*

20. Nama Dialog Screen : Halaman awal admin

Fungsi : Halaman awal untuk admin

Nomor : P20

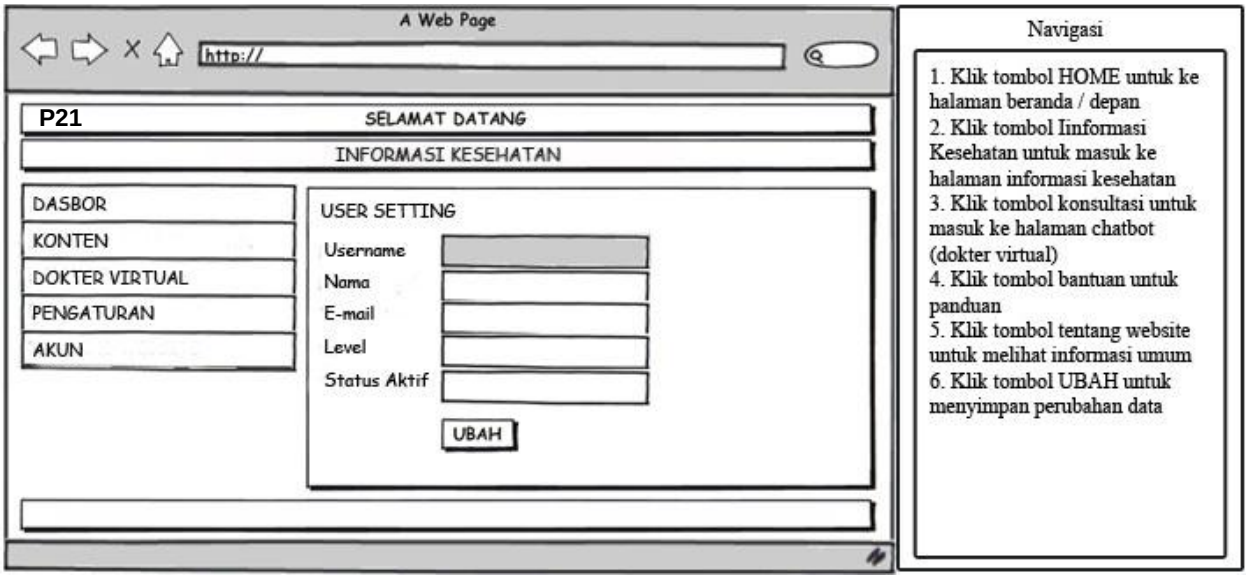


Gambar 3.62 Rancangan Antarmuka Halaman Awal Admin

21. Nama Dialog Screen : Halaman user setting

    Fungsi : Halaman untuk konfigurasi *user setting*

    Nomor : P21

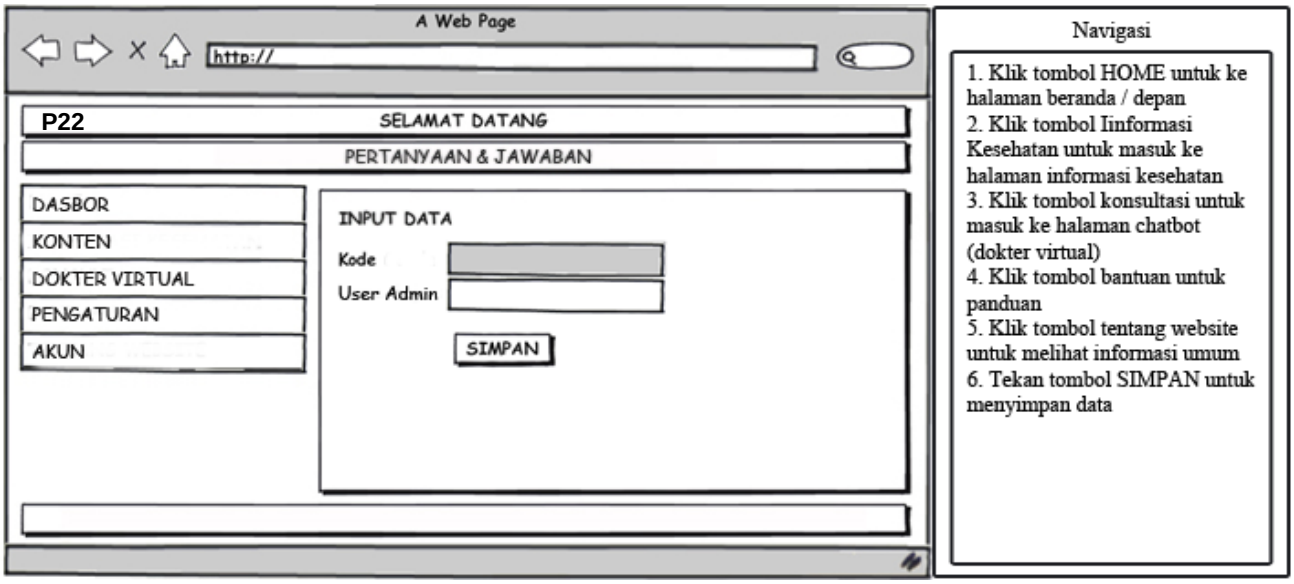


Gambar 3.63 Rancangan Antarmuka Halaman *User Setting*

22. Nama Dialog Screen : Halaman tambah data admin

Fungsi : Halaman untuk menambah data admin

Nomor : P22

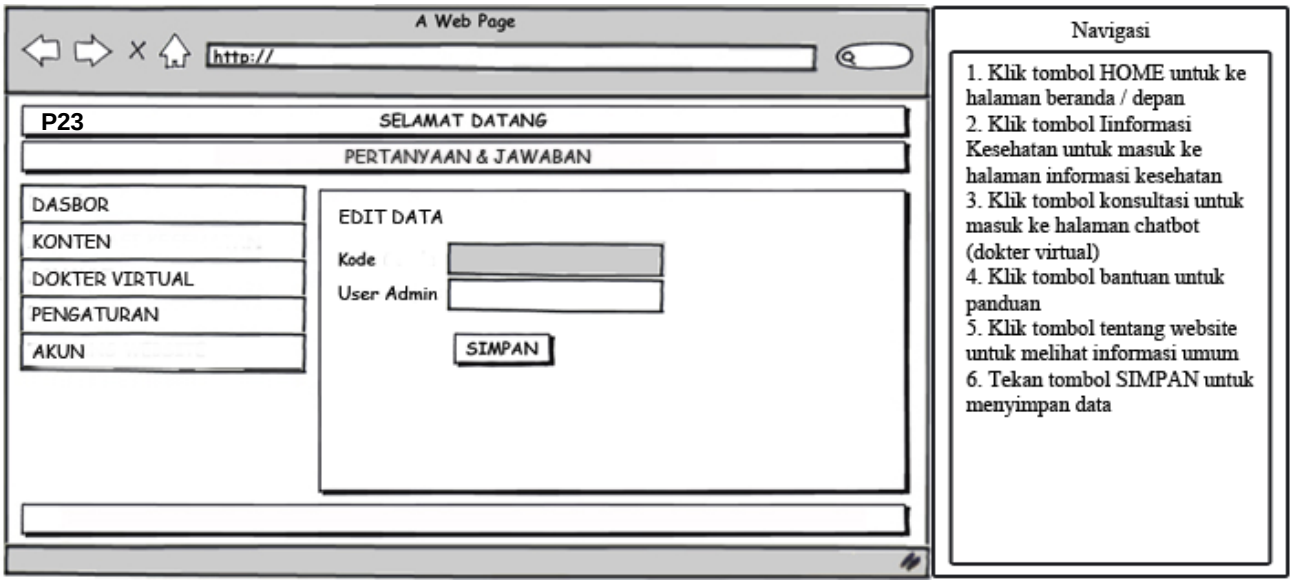


Gambar 3.64 Rancangan antarmuka halaman tambah data admin

23. Nama Dialog Screen : Halaman *edit* data admin

Fungsi : Halaman untuk meng-edit data admin

Nomor : P23

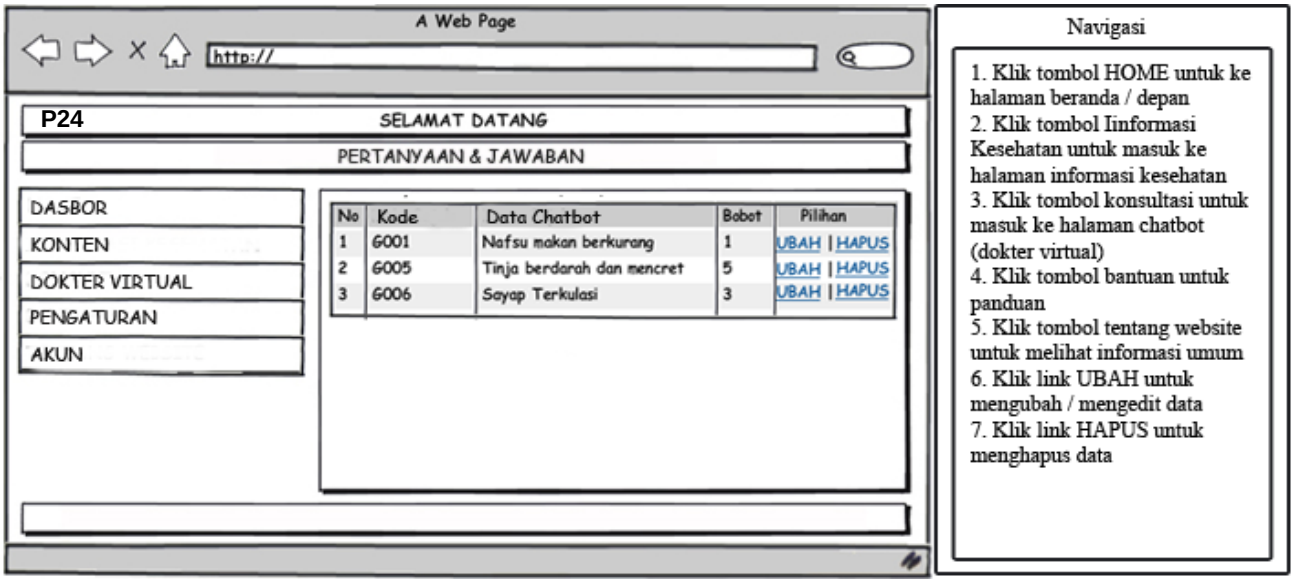


Gambar 3.65 Rancangan antarmuka halaman *edit* data admin

24. Nama Dialog Screen : Halaman data *Chatbot* Dokter *Virtual*

Fungsi : Halaman untuk daftar data *Chatbot* Dokter *Virtual*

Nomor : P24

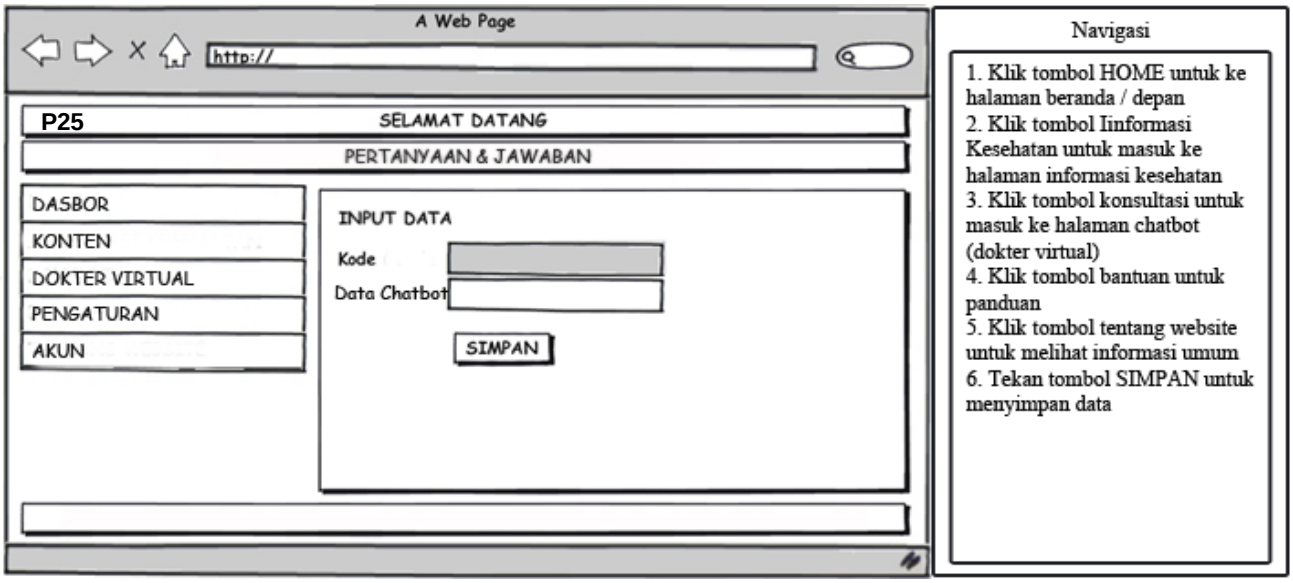


Gambar 3.66 Rancangan Antarmuka Halaman Data *Chatbot* Dokter *Virtual*

25. Nama Dialog Screen : Halaman tambah data *Chatbot* Dokter *Virtual*

Fungsi : Halaman untuk menambah data *Chatbot* Dokter *Virtual*

Nomor : P25

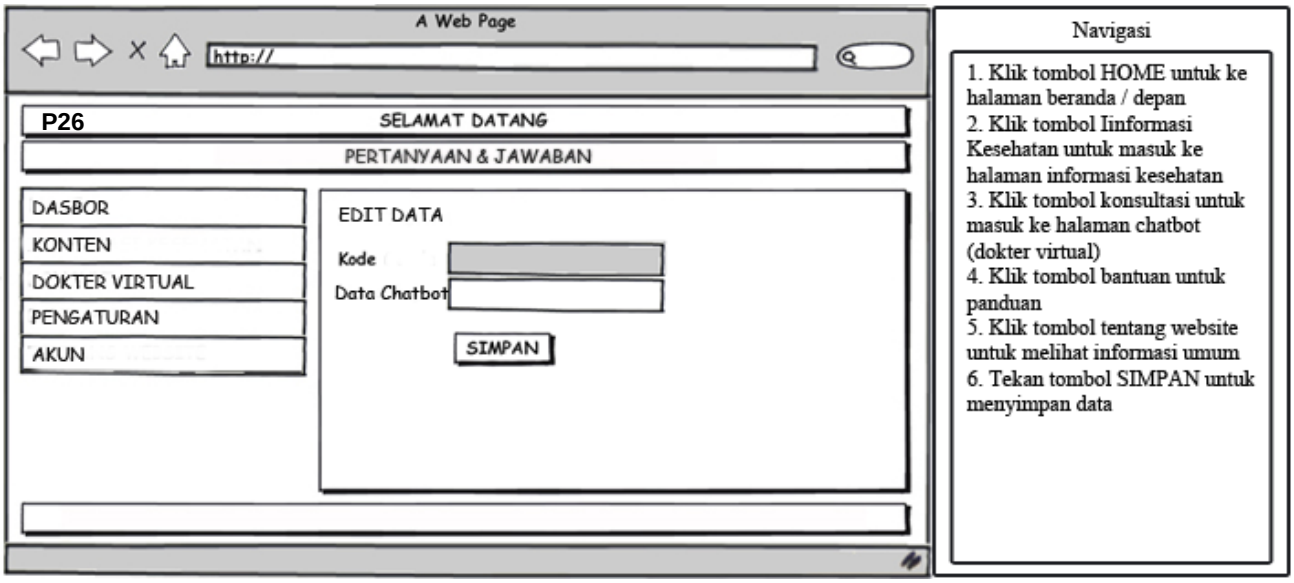


Gambar 3.67 Rancangan antarmuka halaman tambah data chatbot

26. Nama Dialog Screen : Halaman *edit* data *Chatbot* Dokter *Virtual*

Fungsi : Halaman untuk meng-edit data *Chatbot* Dokter *Virtual*

Nomor : P26

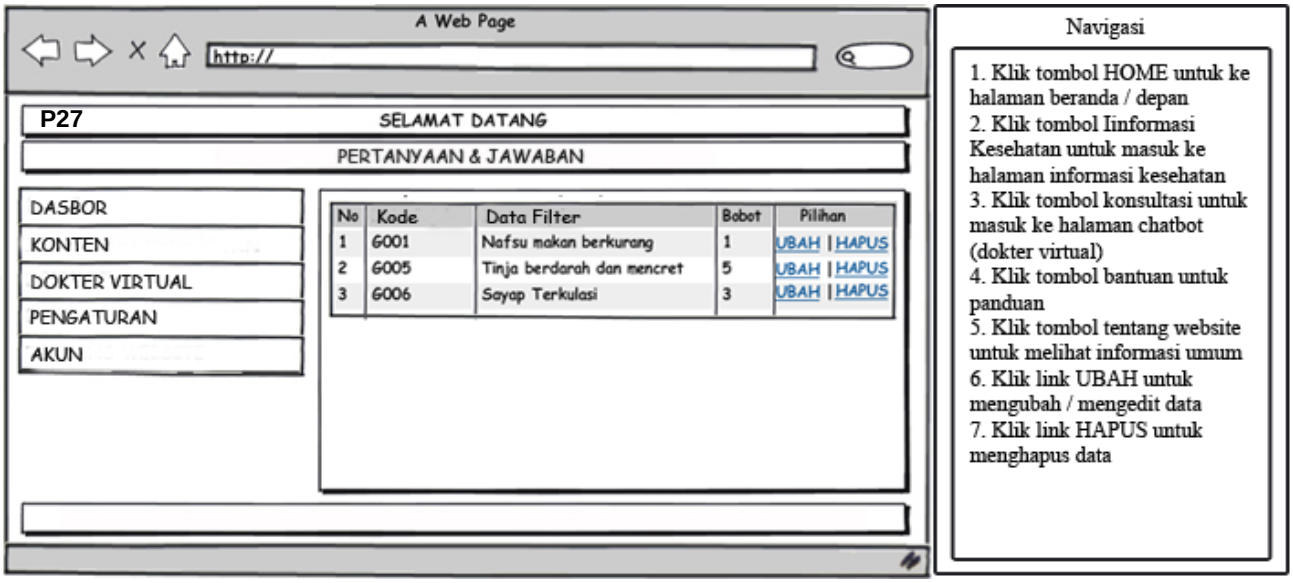


Gambar 3.68 Rancangan antarmuka halaman edit data chatbot

27. Nama Dialog Screen : Halaman data filter

Fungsi : Halaman untuk daftar data filter

Nomor : P27

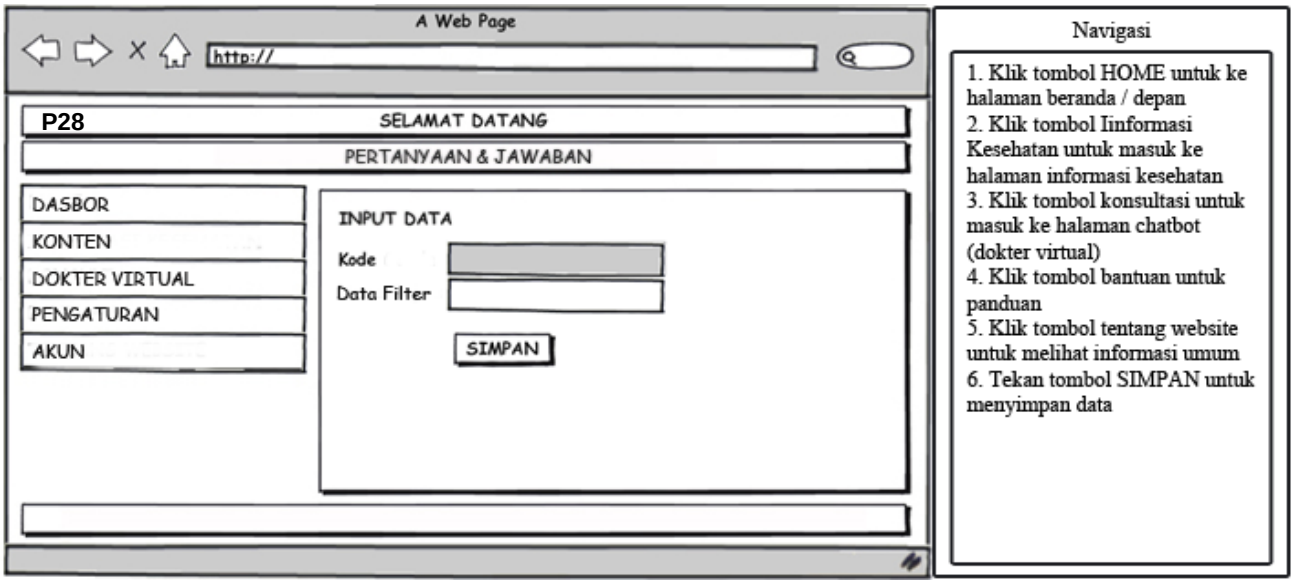


Gambar 3.69 Rancangan Antarmuka Halaman Data Filter

28. Nama Dialog Screen : Halaman tambah data filter

Fungsi : Halaman untuk menambah data filter

Nomor : P28

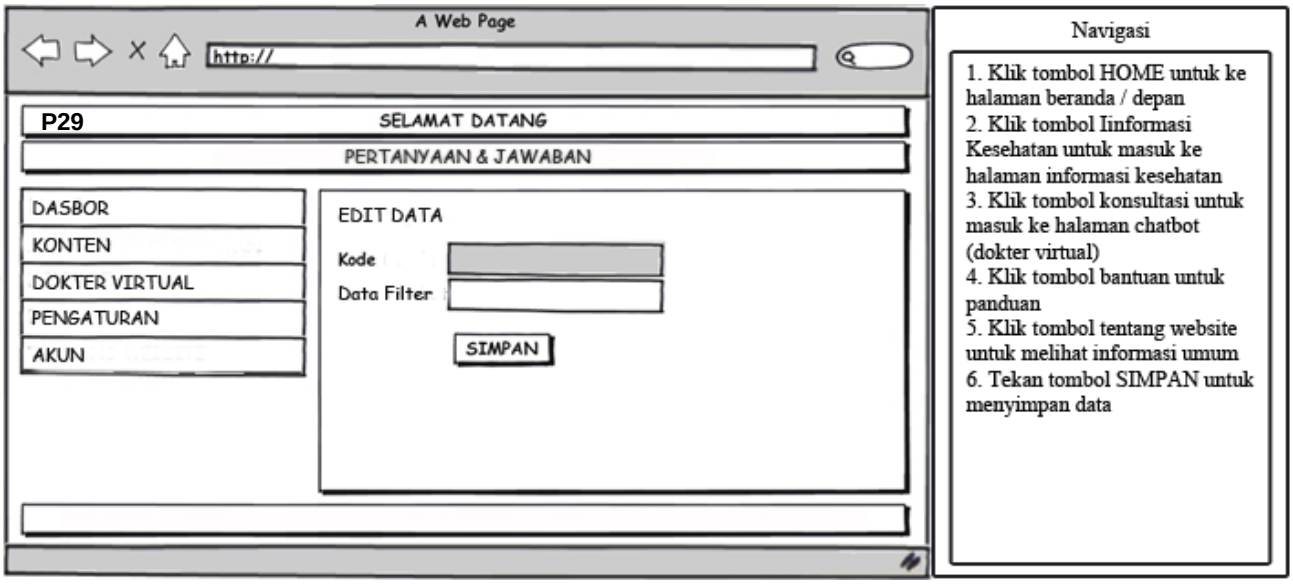


Gambar 3.70 Rancangan antarmuka halaman tambah data filter

29. Nama Dialog Screen : Halaman edit data filter

Fungsi : Halaman untuk meng-*edit* data filter

Nomor : P21

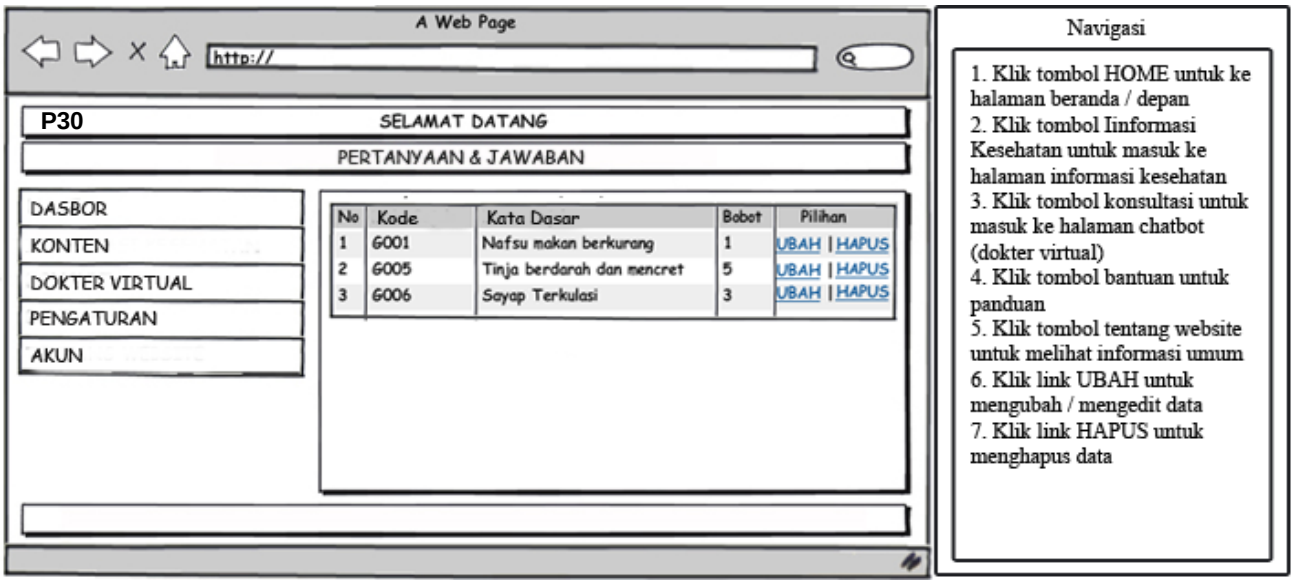


Gambar 3.71 Rancangan antarmuka halaman e*dit* data filter

30. Nama Dialog Screen : Halaman kata dasar

Fungsi : Halaman untuk daftar kata dasar

Nomor : P30

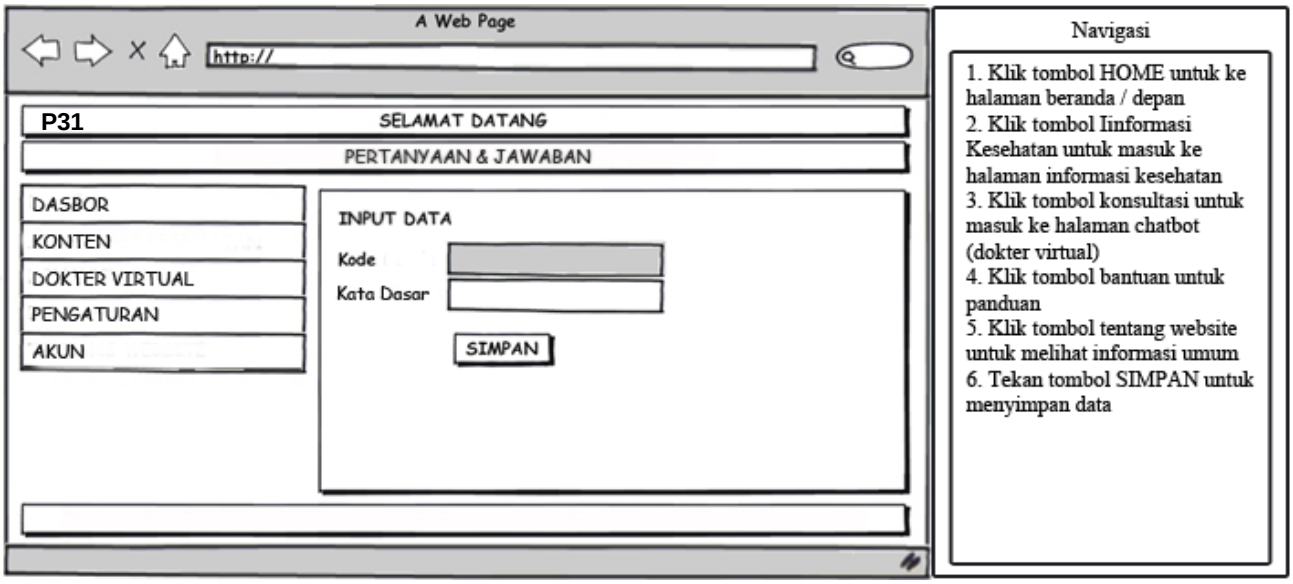


Gambar 3.72 Rancangan Antarmuka Halaman Kata Dasar

31. Nama Dialog Screen : Halaman tambah kata dasar

Fungsi : Halaman untuk menambah kata dasar

Nomor : P31

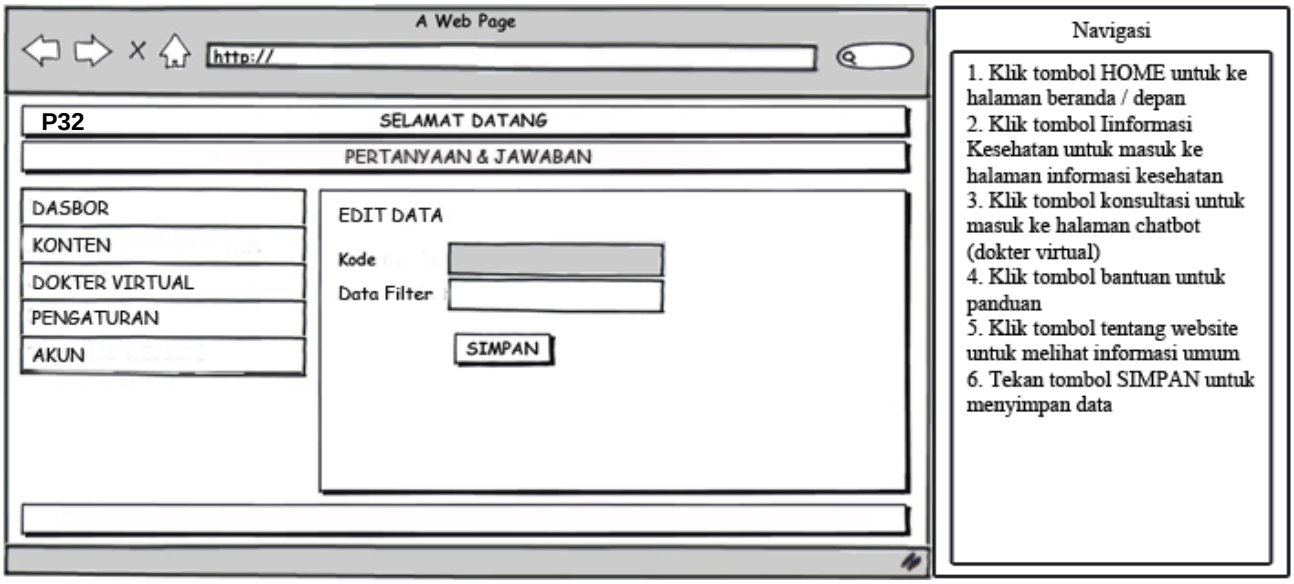


Gambar 3.73 Rancangan antarmuka halaman tambah kata dasar

32. Nama Dialog Screen : Halaman edit kata dasar

Fungsi : Halaman untuk meng-edit kata dasar

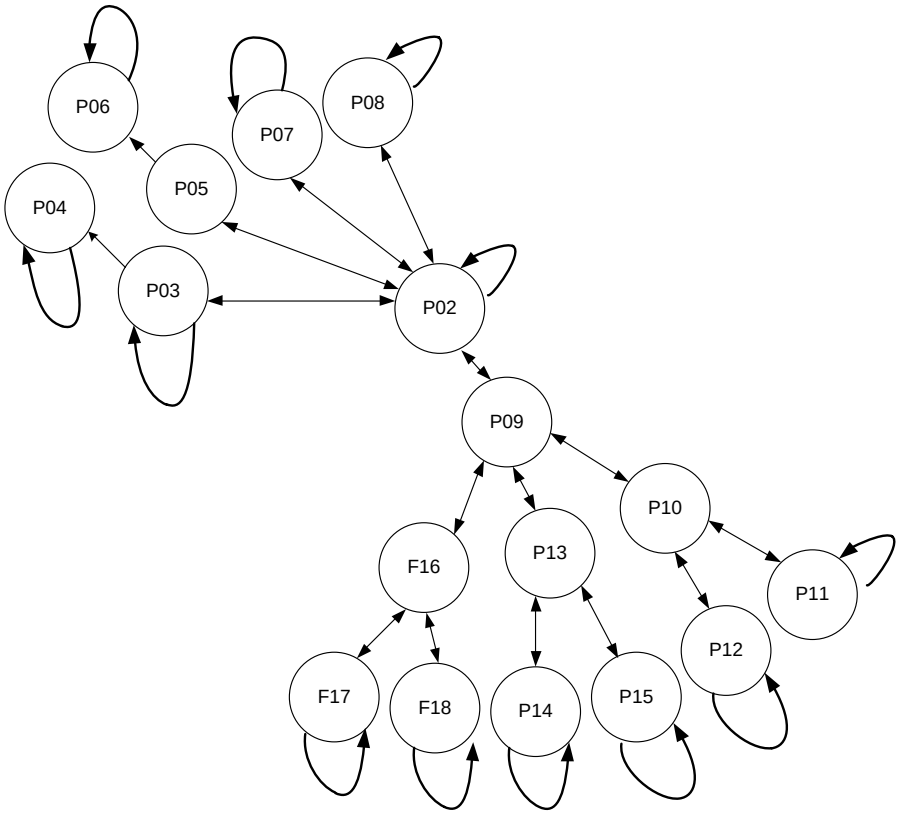
Nomor : P32



Gambar 3.74 Rancangan antarmuka halaman edit kata dasar

3.3.9 Jaringan Semantik

Setelah melakukan perancangan antarmuka dan perancangan pesan maka dilakukanlah perancangan terhadap aliran dari menu-menu yang ada di program akan digambarkan dalam sebuah jaringan *semantik*.



Gambar 3.47 Jaringan Semantik Aplikasi C*hatbot* Dokter *Virtual*