



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB III

METODE DAN PERANCANGAN SISTEM

3.1 Metodologi Penelitian

Langkah-langkah yang dilakukan dalam pembangunan sistem pakar dalam penelitian ini. Penjelasan nya adalah sebagai berikut.

1. Studi Literatur

Studi literatur dilakukan dengan cara mencari informasi secara mandiri mengenai teori dasar yang digunakan dalam penelitian berupa pembangunan suatu sistem pakar untuk mengukur tingkat kesehatan gigi dan mulut berbasis *website*. Pencarian informasi dilakukan melalui sumber buku, jurnal, majalah, dan juga internet. Hal ini dilakukan untuk menambah wawasan dan pengetahuan mengenai topik penelitian agar hasil penelitian bersifat *valid* dan benar adanya, serta untuk mempermudah dalam proses pembangunan sistem pakar.

2. Pengumpulan Data

Data yang dibutuhkan dalam pembangunan program ini adalah data mengenai gejala yang dapat dilihat dan dirasakan secara fisik dan kebiasaan yang dapat mempengaruhi keadaan gigi dan mulut. Data lainnya adalah data riwayat gejala pasien yang pernah memeriksakan keadaan gigi dan mulut ke dokter gigi dan mulut. Pengumpulan data dilakukan dengan cara wawancara langsung dan pengumpulan *form* diagnosis. *Form* diagnosis merupakan *form* yang akan menyimpan hasil diagnosis dari pasien dokter gigi dan mulut berdasarkan gejala yang ada. *Form* tersebut harus diisi berdasarkan sudut pandang dokter gigi dan mulut. *Form* diagnosis berisi pernyataan mengenai gejala dan

kebiasaan pasien serta pernyataan mengenai hasil diagnosis dokter. Seluruh pernyataan yang terdapat pada *form* tersebut telah dikonsultasikan sebelumnya dengan seorang pakar gigi dan mulut.

3. Analisis Model Sistem

Pada tahap ini, dilakukan penentuan gejala dan kebiasaan yang digunakan dalam sistem. Penentuan gejala dan kebiasaan tersebut juga telah dikonsultasikan dengan dokter gigi dan mulut. Selanjutnya, dilakukan analisis model sistem secara terstruktur, yaitu dengan mempertimbangkan keterkaitan data dengan proses yang dapat mengubah data sebagai *entities* terpisah.

4. Perancangan Sistem

Perancangan sistem dimulai dengan merancang antarmuka *website*. Kemudian dilanjutkan dengan perancangan *database* dan perancangan alur sistem. Perancangan sistem dilakukan dengan mengimplementasikan seluruh kebutuhan yang terdapat pada proses analisis model, sehingga seluruh kebutuhan sistem dapat terpenuhi.

5. Pembuatan Program

Program dibangun dengan bahasa pemrograman PHP dan *database* MySQL.

Program yang dibangun merupakan program berbasis *web* yang dapat diakses melalui semua perangkat yang terhubung dengan internet. *Website* dibangun berdasarkan rancangan antarmuka yang telah dibuat pada tahap perancangan sistem. Pada tahap ini, rancangan yang telah dibuat sebelumnya divisualisasikan menggunakan dokumen HTML beserta dengan dokumen CSS yang digunakan untuk membangun tampilan *website* yang menarik, serta menggunakan dokumen jQuery untuk membuat interaksi dengan *website*

menjadi lebih mudah dan efisien. Kemudian dilanjutkan dengan proses penerapan algoritma *forward chaining* dan *dempster shafer* pada program.

6. Pengujian

Pada tahap ini dilakukan pengujian program yang telah dibuat. Pengujian dilakukan mulai dari awal sampai akhir program. Hal ini dilakukan untuk memeriksa apakah ada tampilan atau proses dalam program yang masih mengalami *error*. Selain itu, pengujian juga dilakukan untuk melihat apakah program sudah berjalan sesuai dengan apa yang dirancang dan juga mencari kekurangan dari program agar dapat dikembangkan dan direvisi lebih lanjut.

7. Penarikan Simpulan

Penarikan simpulan merupakan tahapan terakhir dari penelitian. Pada tahap ini dilakukan proses penarikan kesimpulan dan pemberian saran dari sistem yang telah dibuat. Kemudian dilanjutkan dengan proses penulisan laporan mengenai sistem yang telah dibuat.

3.2 Analisis Perancangan

Dalam membangun sistem pakar gigi dan mulut, diperlukan suatu analisis dan perancangan terlebih dahulu. Diagram yang dibuat dalam analisis dan perancangan sistem adalah *Data Flow Diagram (DFD)*, *Entity Relation Diagram (ERD)*, *flowchart*, struktur tabel, dan disain antarmuka (*Interface*).

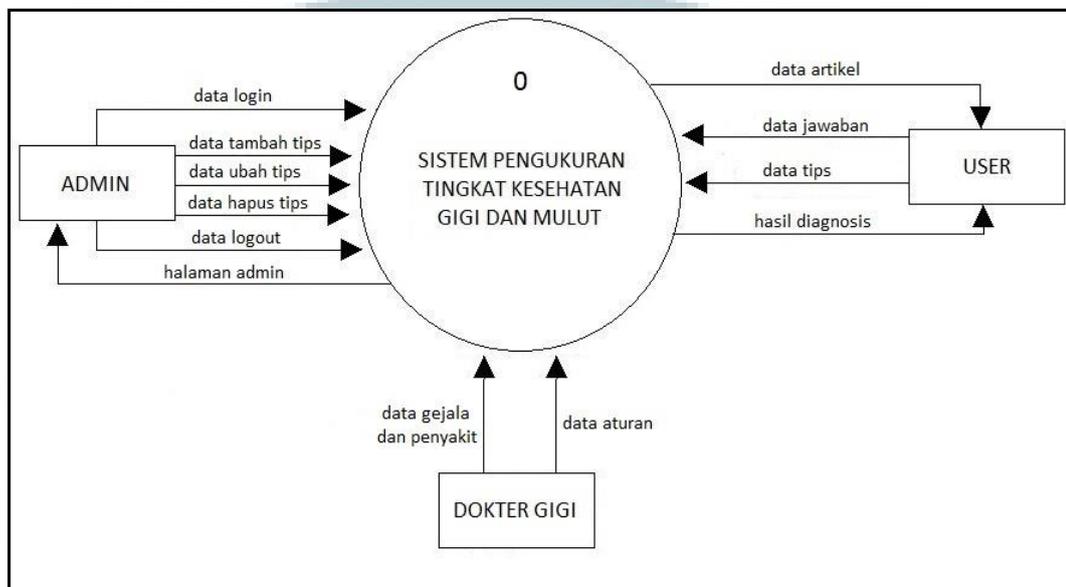
3.2.1 Data Flow Diagram

Data Flow Diagram (DFD) merupakan suatu diagram yang menggunakan notasi-notasi untuk menggambarkan arus dari data sistem, yang penggunaannya sangat membantu untuk memahami sistem secara logika, terstruktur, dan jelas.

Terdapat 3 level di dalam suatu DFD yaitu diagram konteks, diagram nol, dan diagram rinci.

A. Diagram Konteks

Diagram konteks pada sistem digambarkan pada Gambar 3.1.

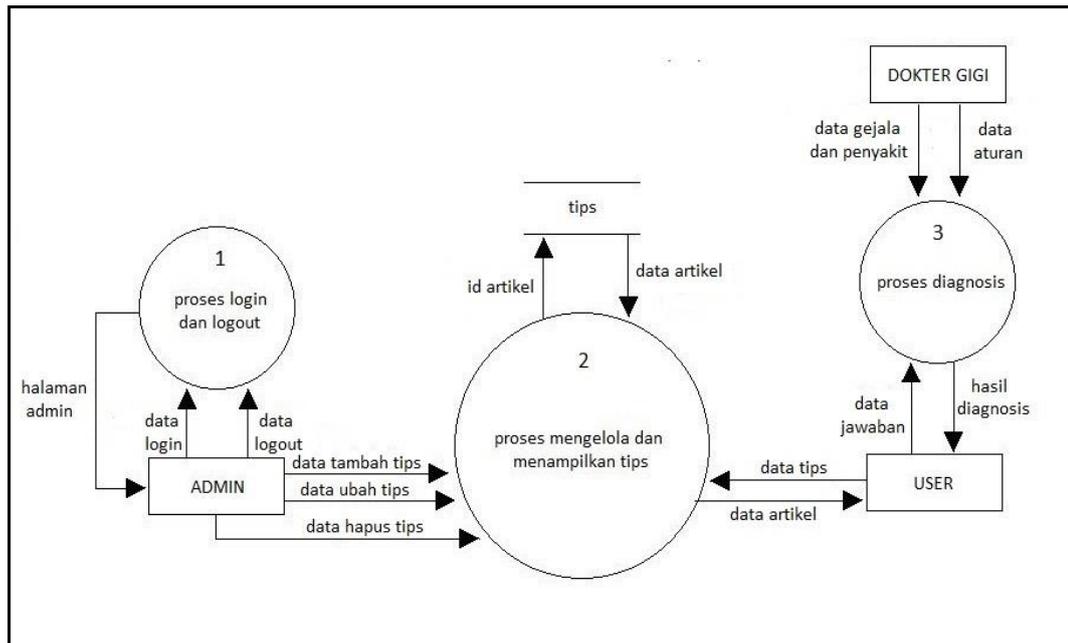


Gambar 3.1 Diagram Konteks

Terdapat tiga entitas eksternal dalam sistem ini, yaitu *admin*, dokter gigi, dan *user*. *Admin* dapat berinteraksi dengan memberikan data *login*, data tambah tips, data ubah tips, data hapus tips, dan data *logout*. Sebaliknya sistem akan memberikan halaman *admin* kepada *admin*. Dokter gigi berinteraksi dengan memberikan data gejala, data penyakit, dan data aturan kepada sistem. *User* dapat berinteraksi dengan sistem dengan memberikan data jawaban dan data tips. Sedangkan sistem akan memberikan hasil diagnosis dan data artikel kepada *user*. Setiap *input* dan *output* yang terjadi pada diagram konteks harus digambarkan kembali pada diagram nol.

B. Diagram Nol

Diagram nol pada sistem digambarkan pada Gambar 3.2.

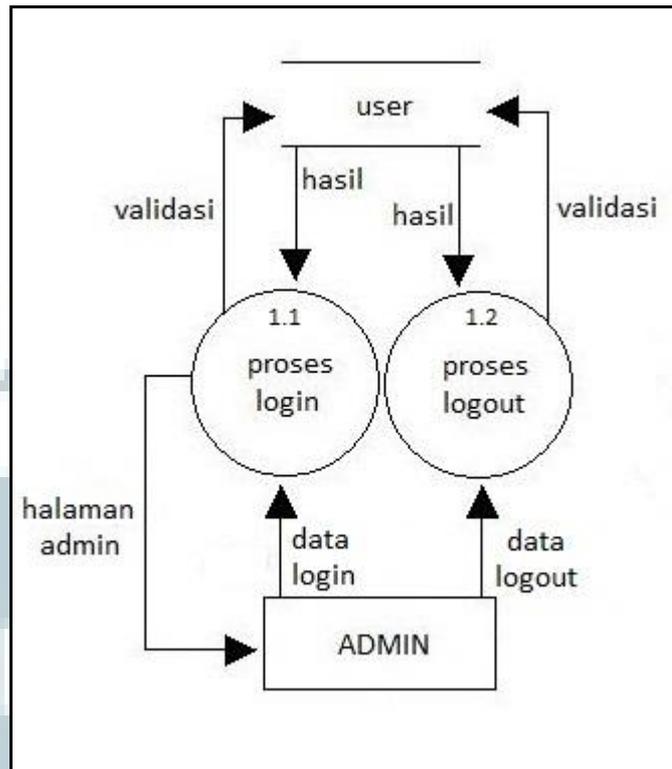


Gambar 3.2 Diagram Nol

Diagram nol merupakan perincian dari diagram konteks. Pada diagram ini, sistem terbagi menjadi tiga proses utama yaitu ‘proses *login* dan *logout*’, ‘proses mengelola dan menampilkan tips’, dan ‘proses diagnosis’. Terdapat pula sebuah *data store* yang terhubung dengan salah satu proses yang ada. Entitas *admin*, dokter gigi, dan *user* memiliki hubungan dengan satu atau lebih proses dari semua proses yang ada. Seluruh proses dan entitas memiliki jumlah *input* dan *output* yang sama dan memiliki nama yang sesuai dengan diagram konteks yang telah dibuat sebelumnya. Semua proses yang masih dapat diperinci akan digambarkan kembali pada diagram rinci.

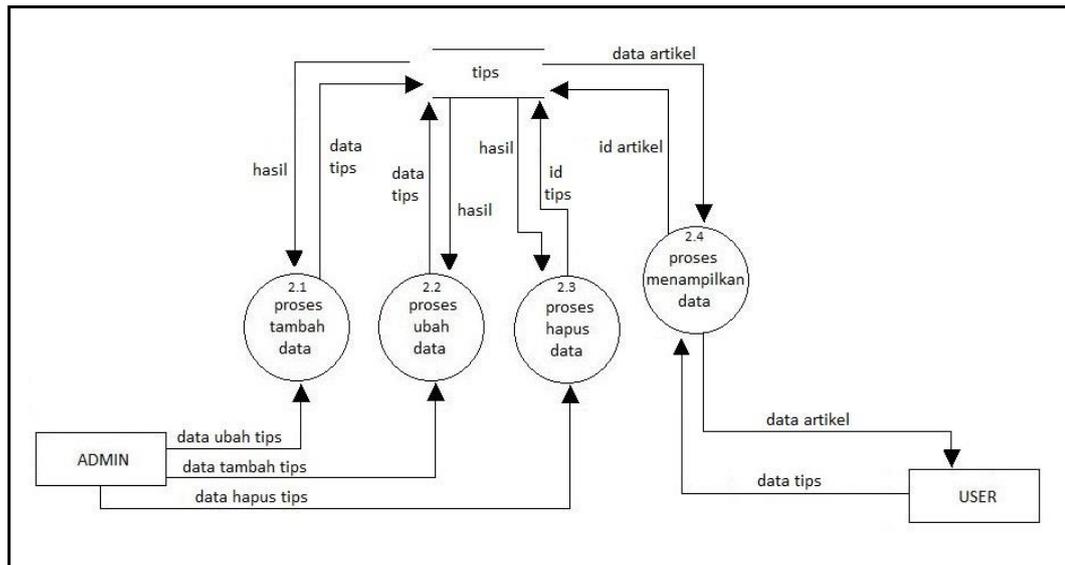
C. Diagram Rinci

Berdasarkan diagram nol pada Gambar 3.2, terdapat tiga proses pada diagram tersebut yang masih dapat diperinci lebih lanjut. Diagram rinci pada sistem digambarkan pada Gambar 3.3, Gambar 3.4, dan Gambar 3.5.



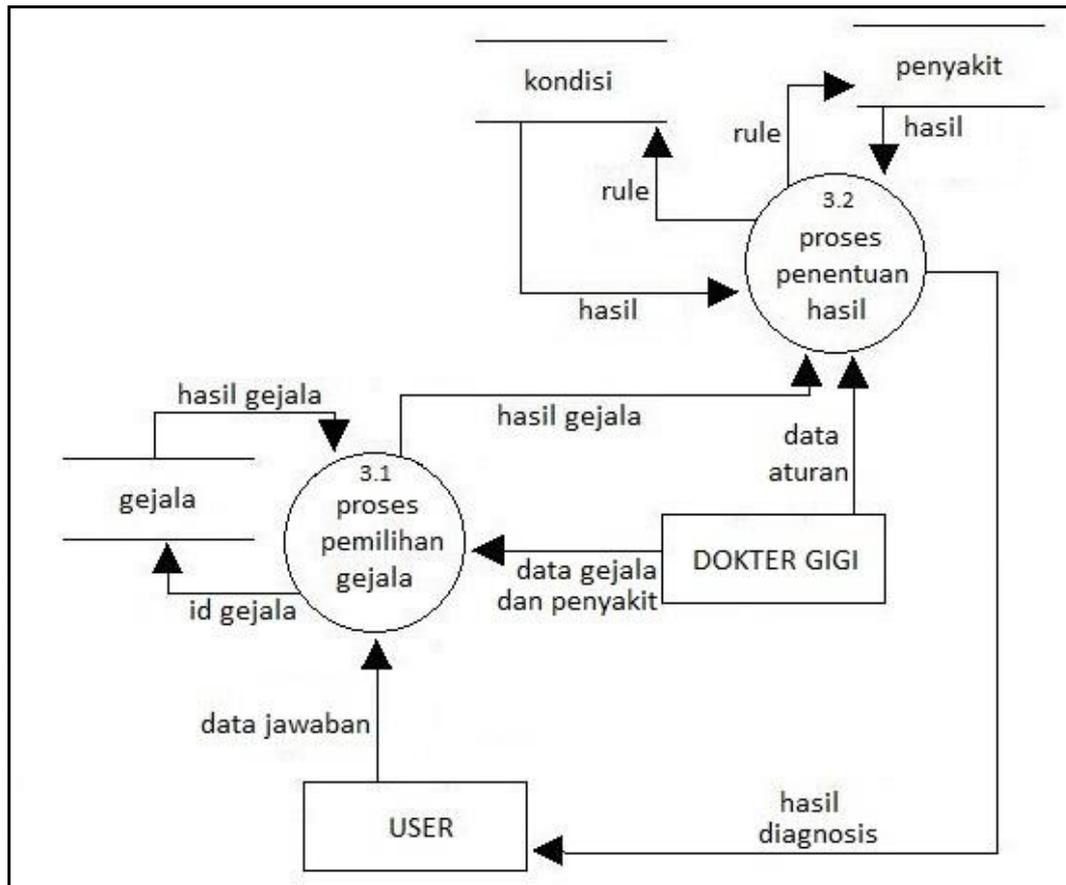
Gambar 3.3 Diagram Rinci Proses *Login* dan *Logout*

Diagram rinci proses *login* dan *logout* melibatkan satu entitas yaitu *admin*. *Admin* memiliki relasi dengan dua proses yang digambarkan di atas. Terdapat dua proses yang digambarkan pada diagram rinci proses *login* dan *logout*, yaitu 'proses login', dan 'proses logout'. Setiap proses terhubung dengan satu *data store*. Hal ini dikarenakan proses membutuhkan data yang terdapat pada *database* sistem. Terdapat sebuah *data store* pada diagram rinci proses *login* dan *logout*, yaitu 'user'. Setiap proses yang ada pada diagram rinci proses *login* dan *logout* memiliki *input* dan *output*.



Gambar 3.4 Diagram Rinci Proses Mengelola dan Menampilkan Tips

Diagram rinci proses mengelola dan menampilkan tips melibatkan dua entitas yaitu *admin* dan *user*. Kedua entitas memiliki relasi dengan satu atau lebih proses yang digambarkan di atas. Terdapat empat proses yang digambarkan pada diagram rinci proses mengelola dan menampilkan tips, yaitu ‘proses tambah data’, ‘proses ubah data’, ‘proses hapus data’, dan ‘proses menampilkan data’. Setiap proses terhubung dengan satu *data store*. Hal ini dikarenakan proses membutuhkan data yang terdapat pada *database* sistem. Terdapat sebuah *data store* pada diagram rinci proses mengelola dan menampilkan tips, yaitu ‘tips’. Setiap proses yang ada pada diagram rinci proses mengelola dan menampilkan tips memiliki *input* dan *output*.



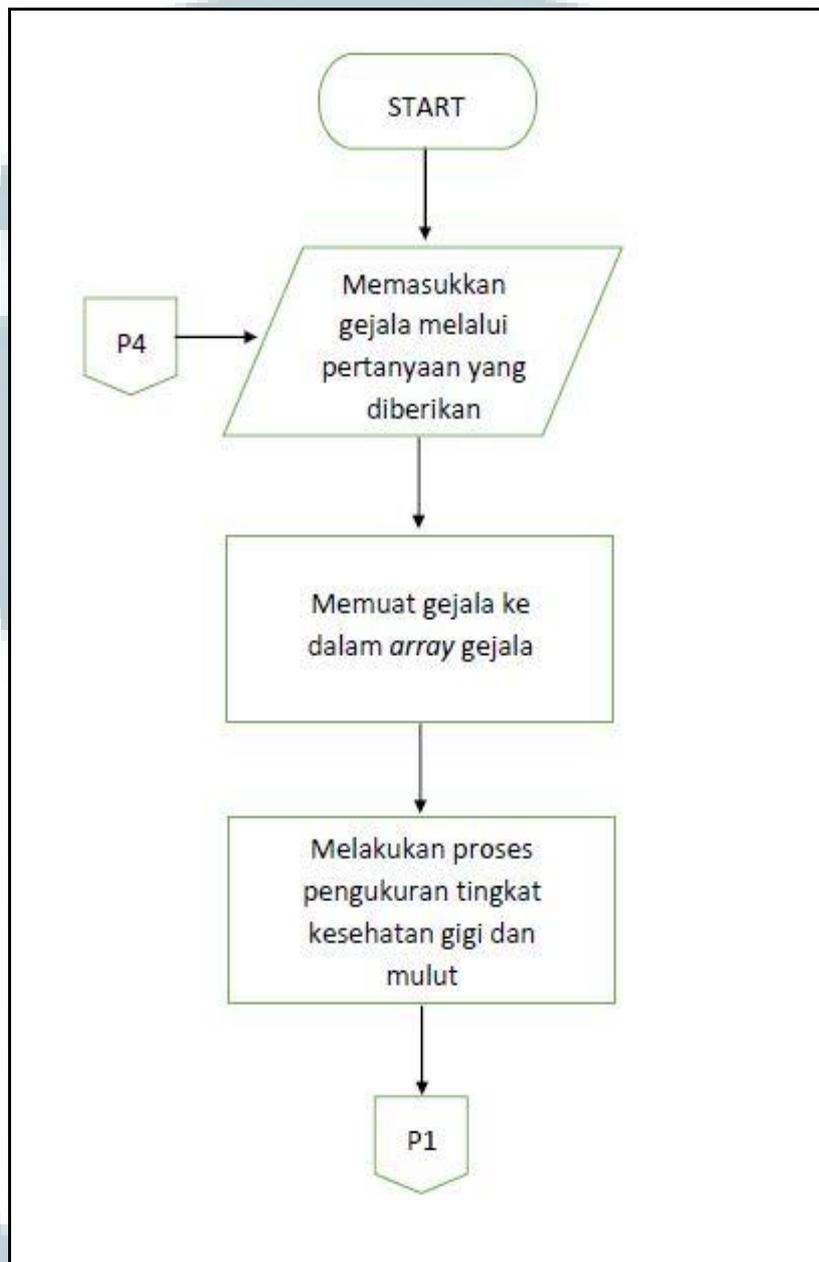
Gambar 3.5 Diagram Rinci Proses Diagnosis

Diagram rinci proses diagnosis melibatkan dua entitas yaitu *user* dan dokter gigi. Kedua entitas memiliki relasi dengan dua proses yang digambarkan di atas. Terdapat dua proses yang digambarkan pada diagram rinci proses diagnosis, yaitu ‘proses pemilihan gejala’, dan ‘proses penentuan hasil’. Setiap proses terhubung dengan satu atau lebih *data store*. Hal ini dikarenakan proses membutuhkan data yang terdapat pada *database* sistem. Terdapat tiga buah *data store* pada diagram rinci proses diagnosis, yaitu ‘gejala’, ‘kondisi’, dan ‘penyakit’. Setiap proses yang ada pada diagram rinci proses diagnosis memiliki *input* dan *output*.

3.2.2 Flowchart

Flowchart merupakan diagram yang menggambarkan hubungan antar proses secara urut dan mendetail yang terjadi pada suatu sistem. Dalam menyusun

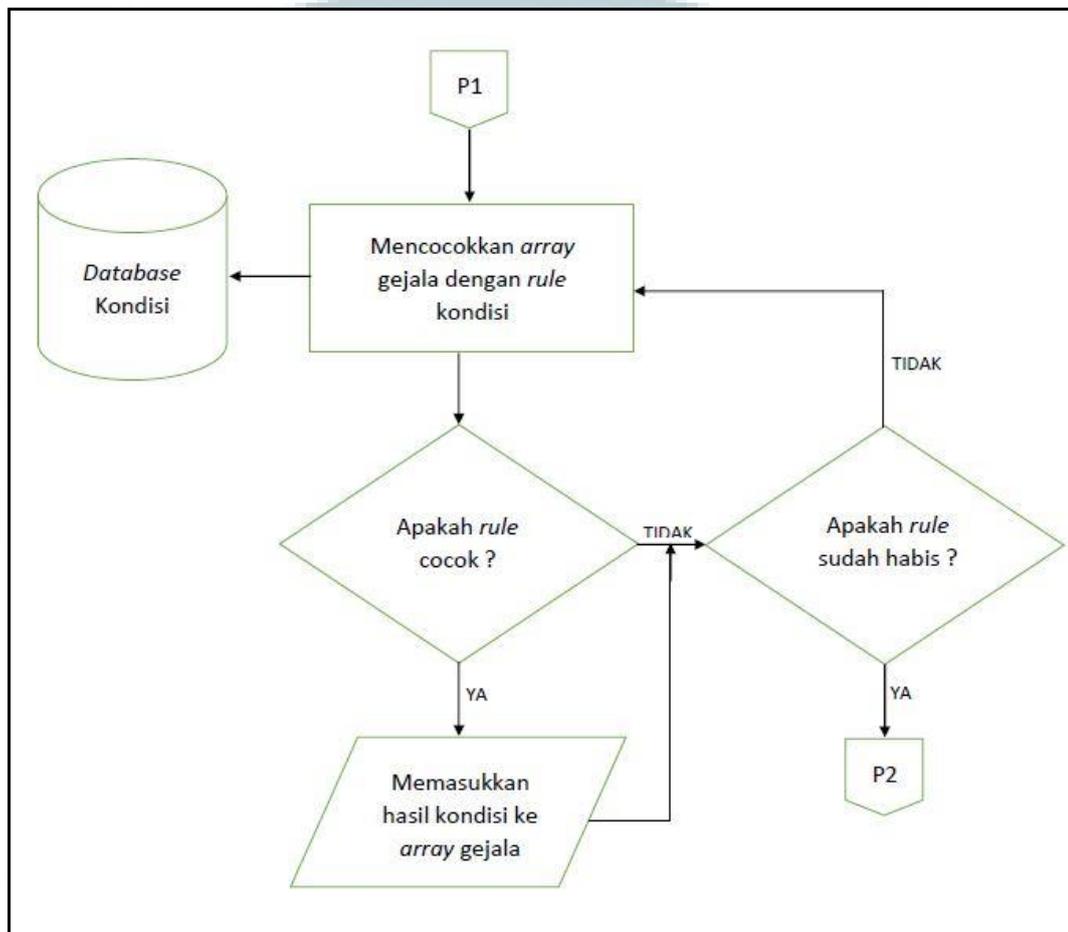
program dari sistem, digunakan dua buah algoritma yaitu *forward chaining* dan *dempster shafer*. *Flowchart* digunakan untuk mempermudah dalam memahami tahapan atau alur kerja dari kedua algoritma tersebut. *Flowchart* sistem dalam melakukan diagnosis digambarkan pada Gambar 3.6.



Gambar 3.6 *Flowchart* Melakukan Diagnosis

Proses dimulai dengan memasukkan gejala dengan cara menjawab pertanyaan yang diberikan oleh sistem. Setelah itu, jawaban akan diproses dan

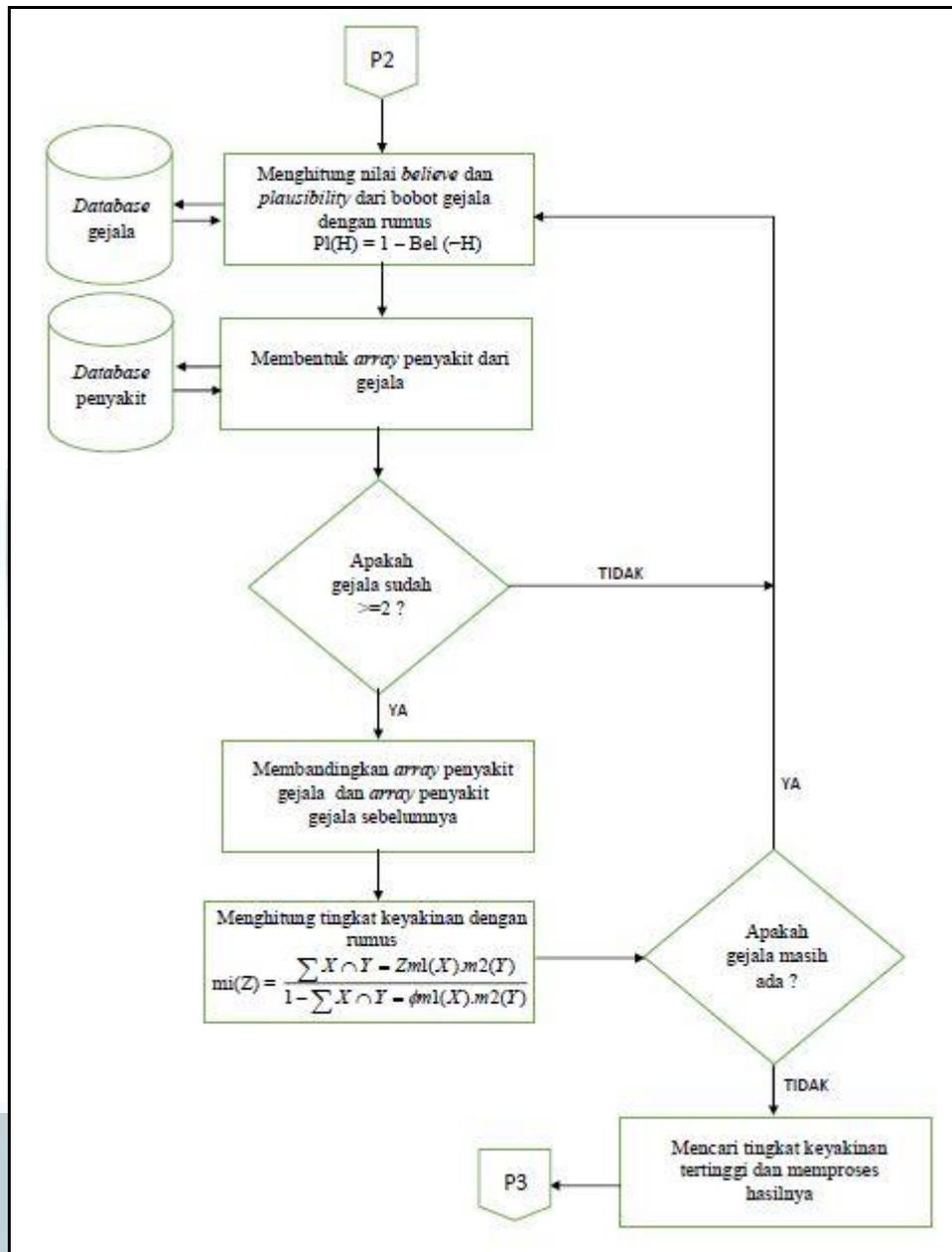
dimasukkan ke dalam *array* gejala. Setelah itu dilakukan proses pengukuran tingkat kesehatan gigi dan mulut. Proses akan dilanjutkan pada P1 yang merupakan proses algoritma *forward chaining*. *Flowchart* algoritma *forward chaining* digambar pada Gambar 3.7.



Gambar 3.7 *Flowchart* Algoritma *Forward Chaining*

Proses algoritma *forward chaining* akan dimulai dengan mencocokkan antara gejala yang dimasukkan dengan *rule* atau aturan dari *database* kondisi. Apabila *rule* cocok, maka hasil *rule* akan dimasukkan ke dalam *array* gejala. Kemudian akan dilakukan pengecekan apakah *rule* sudah habis atau belum. Apabila *rule* sudah habis maka proses akan dilanjutkan ke P2. Apabila *rule* belum habis maka langkah diulang kembali ke pencocokan gejala dengan *rule*. P2 merupakan proses

algoritma *dempster shafer*. Flowchart yang menunjukkan cara kerja algoritma *dempster shafer* digambarkan pada Gambar 3.8.

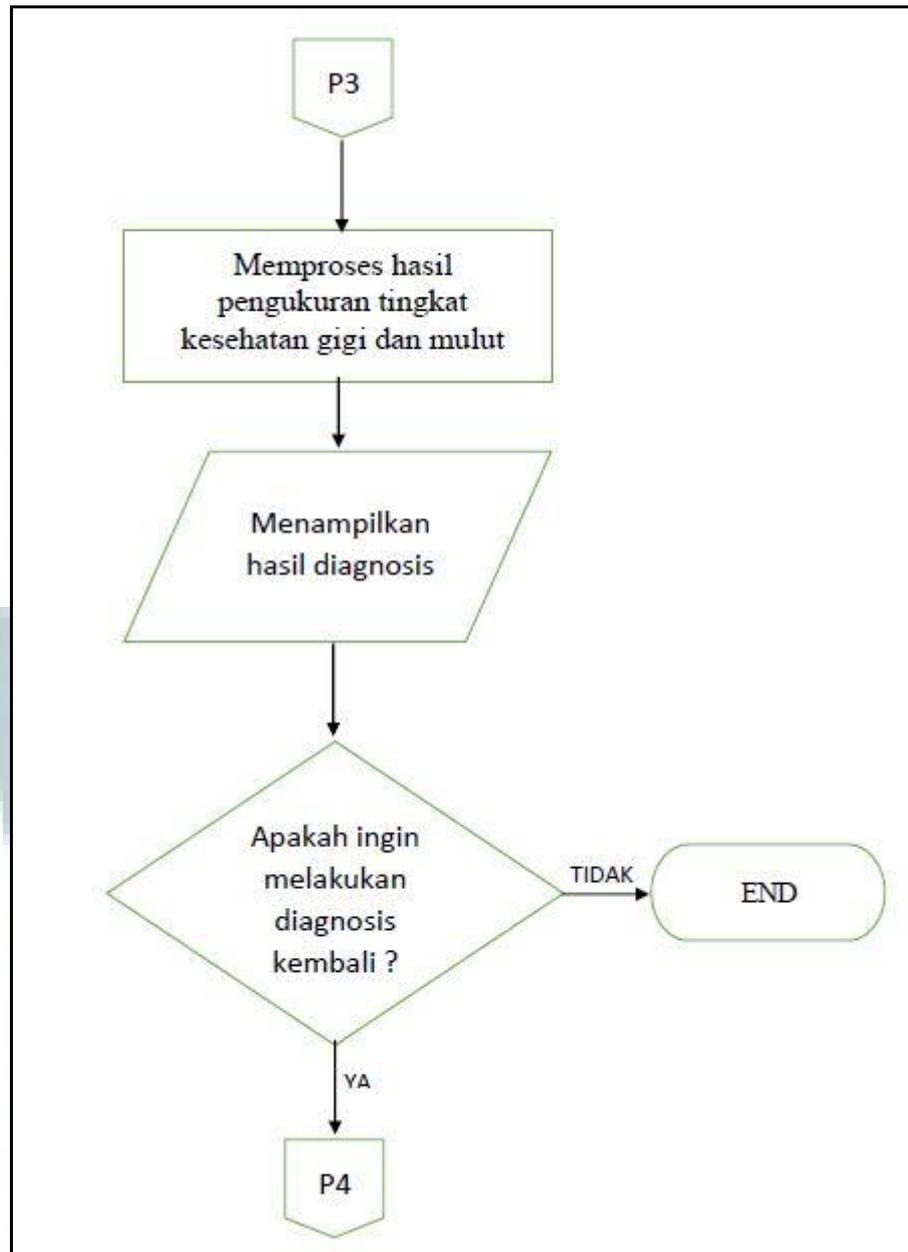


Gambar 3.8 Flowchart Algoritma Dempster Shafer

Algoritma *dempster shafer* dimulai dengan melakukan perhitungan nilai *believe* dan nilai *plausibility* dengan rumus yang telah ditentukan berdasarkan nilai bobot gejala pada *database* gejala. Kemudian akan dilakukan pembentukan *array* penyakit berdasarkan gejala tersebut. Setelah itu, akan dilakukan

pengecekan apakah gejala yang telah diproses berjumlah lebih besar atau sama dengan dua. Apabila sudah lebih besar atau sama dengan dua maka akan dilakukan perbandingan *array* penyakit dari gejala dengan *array* penyakit dari gejala yang telah diproses sebelumnya. Setelah itu, akan dilakukan perhitungan nilai tingkat keyakinan dengan rumus yang telah ditentukan. Kemudian akan dilakukan pengecekan apakah gejala masih ada atau tidak. Apabila sudah tidak ada maka proses akan dilanjutkan ke pencarian tingkat keyakinan tertinggi kemudian dilanjutkan ke proses P4. Apabila gejala masih belum habis maka proses akan kembali ke perhitungan nilai *believe* dan nilai *plausibility* gejala. Proses P4 merupakan proses yang menampilkan hasil dari diagnosis sistem. *Flowchart* menampilkan hasil diagnosis digambarkan pada Gambar 3.9.

UMMN



Gambar 3.9 *Flowchart* Menampilkan Hasil Diagnosis

Proses menampilkan hasil diagnosis dimulai dengan memproses hasil pengukuran tingkat kesehatan gigi dan mulut. Setelah itu akan ditampilkan hasil diagnosis. *User* mendapatkan pilihan apakah ingin melakukan proses diagnosis kembali. Apabila *user* ingin melakukan proses diagnosis kembali, maka proses akan dilanjutkan ke P4 yang akan dilanjutkan dengan proses memasukkan gejala dengan cara menjawab pertanyaan dari sistem.

3.2.3 Struktur Tabel

Terdapat 5 tabel yang tidak berelasi. Kelima struktur tabel yang digunakan dalam pembuatan sistem ini adalah sebagai berikut.

- 1) Nama Tabel : Gejala
Fungsi : Menyimpan data gejala
Primary Key : *id*
Foreign Key : -

Struktur dari tabel gejala dapat dilihat pada Tabel 3.1.

Tabel 3.1 Struktur Tabel Gejala

Nama Kolom	Tipe	Panjang	Keterangan
Id	Int	11	primary key
Nama	Varchar	100	
Status	Int	11	
created_at	Datetime		
updated_at	Datetime		
Bobot	Double		

- 2) Nama Tabel : Kondisi
Fungsi : Menyimpan *rule* untuk menentukan kondisi mulut
Primary Key : *id*
Foreign Key : -

Struktur dari tabel kondisi dapat dilihat pada Tabel 3.2.

Tabel 3.2 Struktur Tabel Kondisi

Nama Kolom	Tipe	Panjang	Keterangan
Id	Int	11	primary key
Goal	Varchar	20	
Gejala_id	Varchar	100	

- 3) Nama Tabel : Penyakit
Fungsi : Menyimpan *rule* untuk menentukan penyakit

Primary Key : *id*

Foreign Key : -

Struktur dari tabel penyakit dapat dilihat pada Tabel 3.3.

Tabel 3.3 Struktur Tabel Penyakit

Nama Kolom	Tipe	Panjang	Keterangan
Id	Int	11	primary key
Nama	Varchar	100	
Gejala_id	Varchar	300	
Solusi	Varchar	1000	

4) Nama Tabel : *Tips*

Fungsi : Menyimpan data tentang tips

Primary Key : *id*

Foreign Key : -

Struktur dari tabel tips dapat dilihat pada Tabel 3.4.

Tabel 3.4 Struktur Tabel Tips

Nama Kolom	Tipe	Panjang	Keterangan
Id	Int	5	primary key
Gambar	Varchar	100	
Judul	Varchar	100	
Berita_pendek	Varchar	100	
Berita_panjang	Varchar	10000	
Sumber	Varchar	100	
Created_at	Datetime		

5) Nama Tabel : *User*

Fungsi : Menyimpan data admin

Primary Key : *id*

Foreign Key : -

Struktur dari tabel *user* dapat dilihat pada Tabel 3.5.

Tabel 3.5 Struktur Tabel User

Nama Kolom	Tipe	Panjang	Keterangan
Id	Int	2	primary key
Username	Varchar	60	
Password	Varchar	60	
Remember_token	Varchar	60	
Created_at	Datetime		
Updated_at	Datetime		

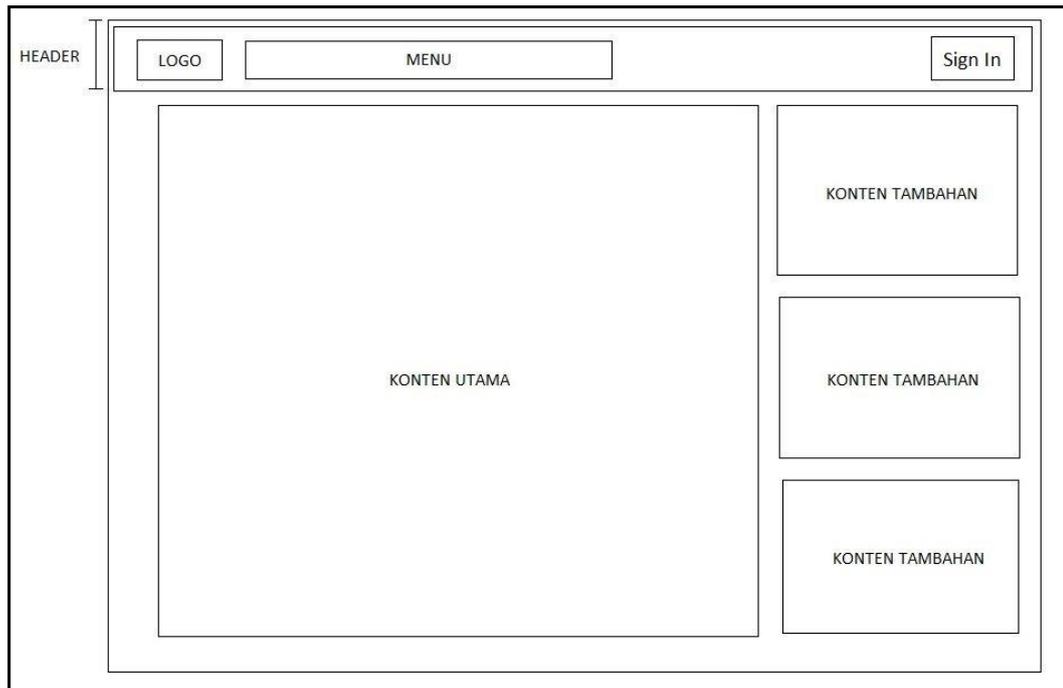
3.2.4 Perancangan Antarmuka Website

Sistem pakar dalam penelitian ini dibuat dalam bentuk *website*, sehingga dapat diakses melalui semua perangkat yang dapat terhubung langsung dengan internet. Seluruh rancangan *website* ini dirancang agar dapat tampil *responsive* pada segala resolusi layar menggunakan *framework bootstrap*. Rancangan antarmuka utama *website* ini terbagi menjadi dua bagian yaitu rancangan halaman muka (*frontend*) dan rancangan halaman admin (*backend*).

A. Rancangan Halaman Muka

Halaman muka (*frontend*) ditujukan untuk *user* atau pengguna yang mengakses *website*. *User* tidak perlu melakukan *login* untuk dapat melihat tampilan *frontend*.

Kerangka utama *frontend* terdiri atas tiga bagian utama yaitu bagian *header* yang memiliki logo, menu, dan tombol *sign in* untuk *admin*, bagian konten utama yang merupakan *container* dari konten utama *website*, dan bagian konten tambahan yang terletak di sebelah kanan konten utama. Apabila layar diperkecil seukuran layar *smartphone* maka *header* hanya akan terdiri dari logo dan *toggle button* yang apabila ditekan akan memunculkan menu dan tombol *sign in* untuk *admin*. Konten tambahan akan berada di bawah konten utama. Kerangka utama dari halaman *frontend* dapat dilihat pada Gambar 3.10.

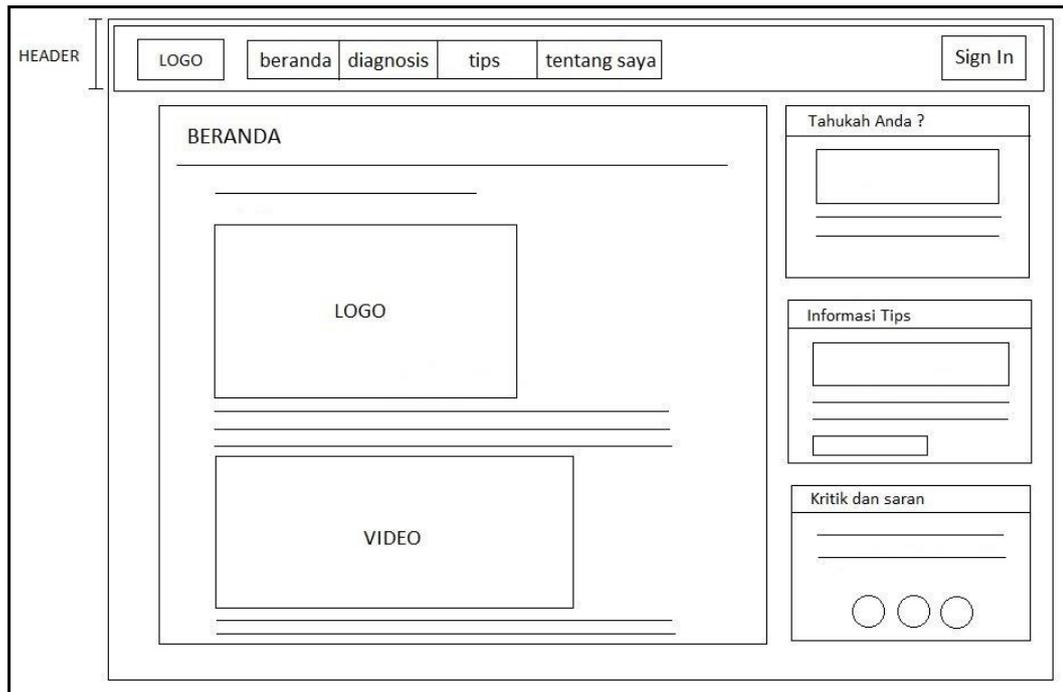


Gambar 3.10 Rancangan Halaman *Frontend*

Halaman *frontend website* memiliki empat menu utama yaitu beranda, diagnosis, tips, tentang saya, dan *login*. Oleh karena itu, *frontend* terdiri dari lima halaman utama yang akan dijelaskan sebagai berikut.

1. Halaman Beranda

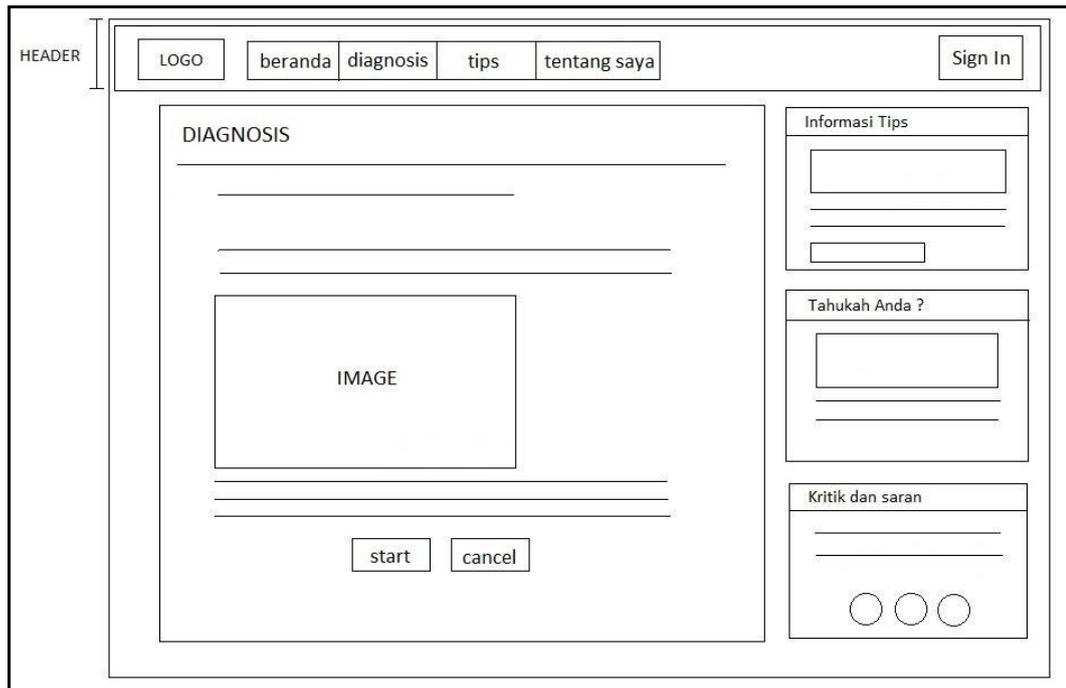
Halaman beranda merupakan halaman utama dari *frontend website*. Halaman ini memiliki tiga bagian utama, yaitu *header*, konten utama, dan konten tambahan. Bagian *header* memiliki logo, menu, dan tombol *sign in*. Pada bagian konten utama terdapat logo *website*, *video* tutorial, dan terdapat penjelasan mengenai kegunaan *website* dan *link* ke menu-menu lainnya. Bagian konten tambahan berisikan info sederhana, artikel tips yang dapat langsung terhubung ke halaman artikel, dan kritik dan saran yang dapat dikirimkan melalui tombol lingkaran yang sudah terhubung langsung dengan media sosial. Untuk lebih jelasnya, halaman beranda digambarkan pada Gambar 3.11.



Gambar 3.11 Halaman Beranda

2. Halaman Diagnosis

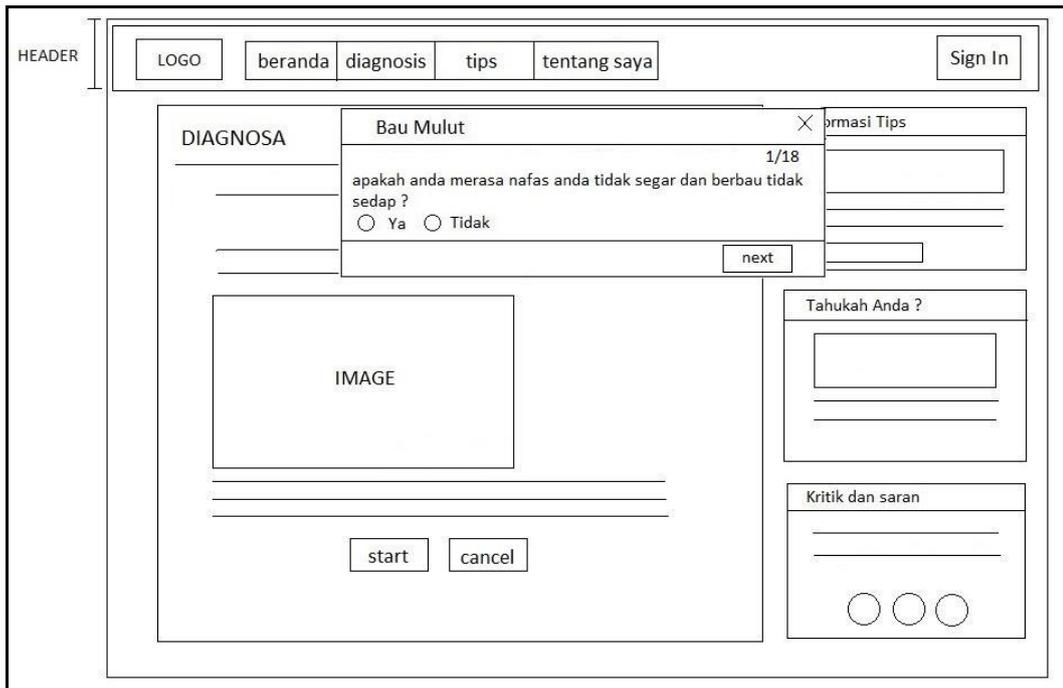
Halaman diagnosis merupakan bagian utama dari *frontend* website. Halaman ini memiliki tiga bagian utama, yaitu *header*, konten utama, dan konten tambahan. Bagian *header* memiliki logo, menu, dan tombol *sign in*. Pada bagian konten utama terdapat sebuah *image*, penjelasan singkat, dan terdapat tombol *start* dan *cancel* yang akan digunakan untuk memulai atau membatalkan proses diagnosis. Bagian konten tambahan berisikan info sederhana, artikel tips yang dapat langsung terhubung ke halaman artikel, dan kritik dan saran yang dapat dikirimkan melalui tombol lingkaran yang sudah terhubung langsung dengan media sosial. Tampilan halaman diagnosis digambarkan pada Gambar 3.12.



Gambar 3.12 Halaman Diagnosis

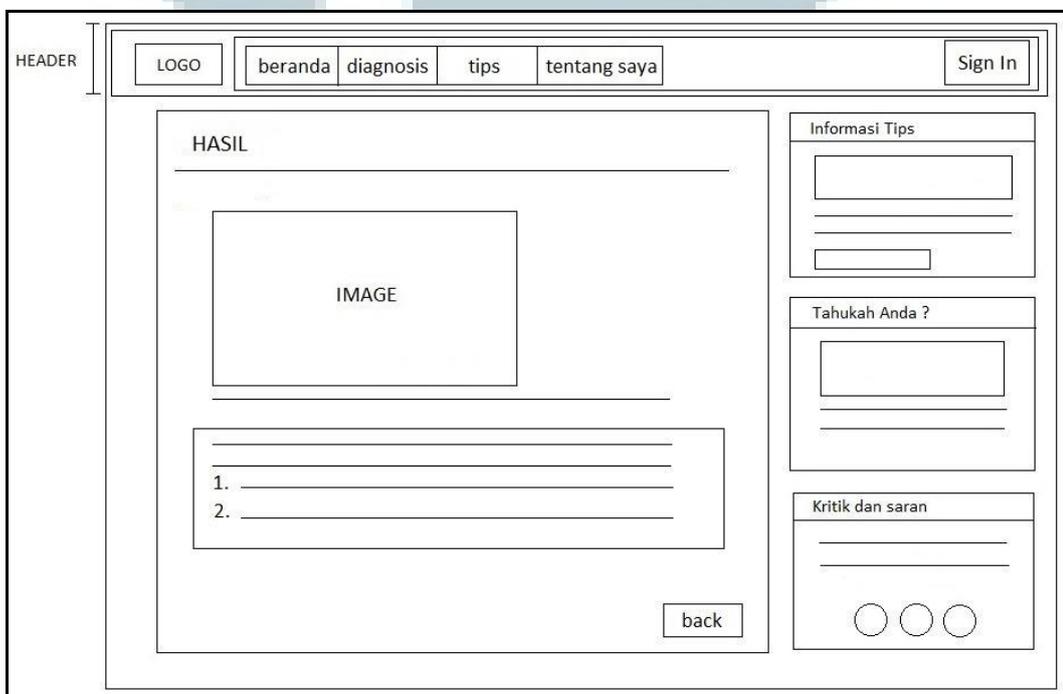
Apabila tombol *start* ditekan maka akan muncul *modal* yang berisikan pertanyaan. Setelah memilih jawaban, *user* kemudian dapat menekan tombol *next* untuk melanjutkan ke pertanyaan berikutnya. Namun, *user* tidak dapat melanjutkan ke pertanyaan berikutnya apabila belum menjawab pertanyaan yang ada. Tombol *next* akan bersifat *disable* apabila jawaban belum ditentukan. Untuk lebih jelasnya, tampilan pertanyaan berupa *modal* akan digambarkan pada Gambar 3.13.

U
M
M
N



Gambar 3.13 Tampilan *Modal*

Setelah seluruh pertanyaan selesai di jawab, maka *user* dapat menekan tombol *finish* untuk mengetahui hasil yang didapat pada halaman hasil. Tampilan halaman hasil digambarkan pada Gambar 3.14.



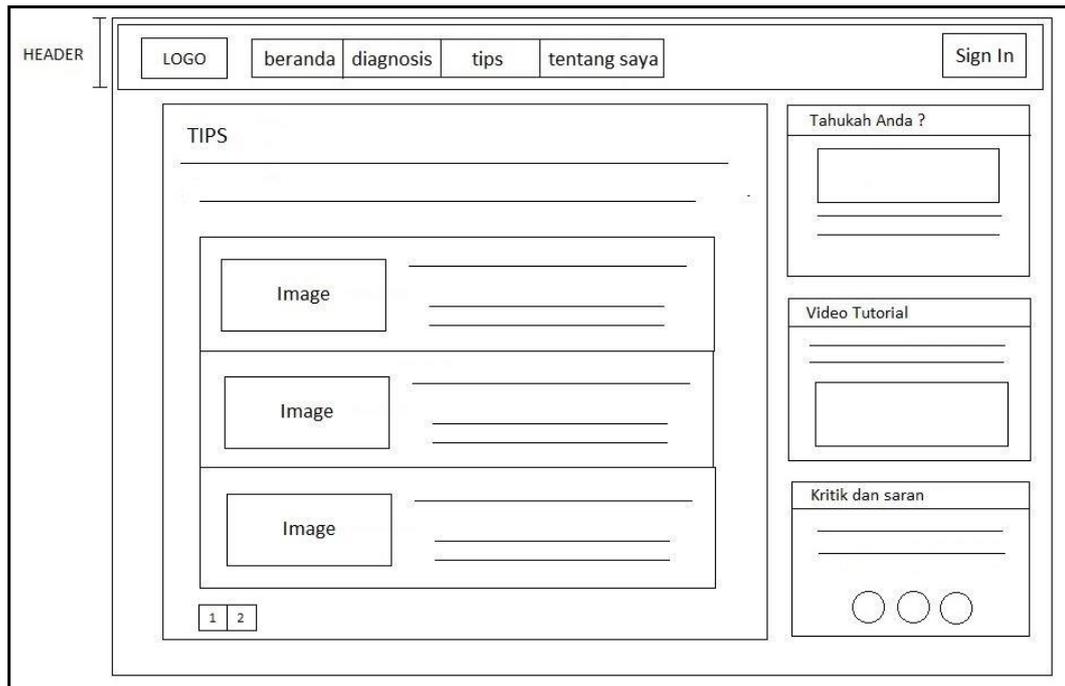
Gambar 3.14 Halaman Hasil

Bagian konten utama dari halaman hasil adalah sebuah *image*, hasil diagnosis tentang tingkat kesehatan gigi dan mulut (sehat, kurang sehat, atau tidak sehat), serta hasil diagnosis penyakit yang mungkin menyerang dengan kemungkinan dalam persen. Hasil diagnosis penyakit yang ditampilkan bisa saja berjumlah lebih dari satu jenis penyakit tergantung pada hasil perhitungan dari algoritma.

3. Halaman Tips

Halaman tips merupakan bagian utama dari *frontend* website. Halaman ini berisikan artikel tentang kesehatan gigi dan mulut. Halaman ini memiliki tiga bagian utama, yaitu *header*, konten utama, dan konten tambahan. Bagian *header* memiliki logo, menu, dan tombol *sign in*. Pada bagian konten utama terdapat sebuah *listview* yang berisikan konten berita seperti *image*, judul berita, dan sedikit berita yang ditampilkan. *Listview* ini dapat ditekan dan langsung terhubung dengan halaman artikel. Bagian konten tambahan berisikan info sederhana, *video* tutorial, dan kritik dan saran yang dapat dikirimkan melalui tombol lingkaran yang sudah terhubung langsung dengan media sosial. Tampilan halaman tips digambarkan pada Gambar 3.15.

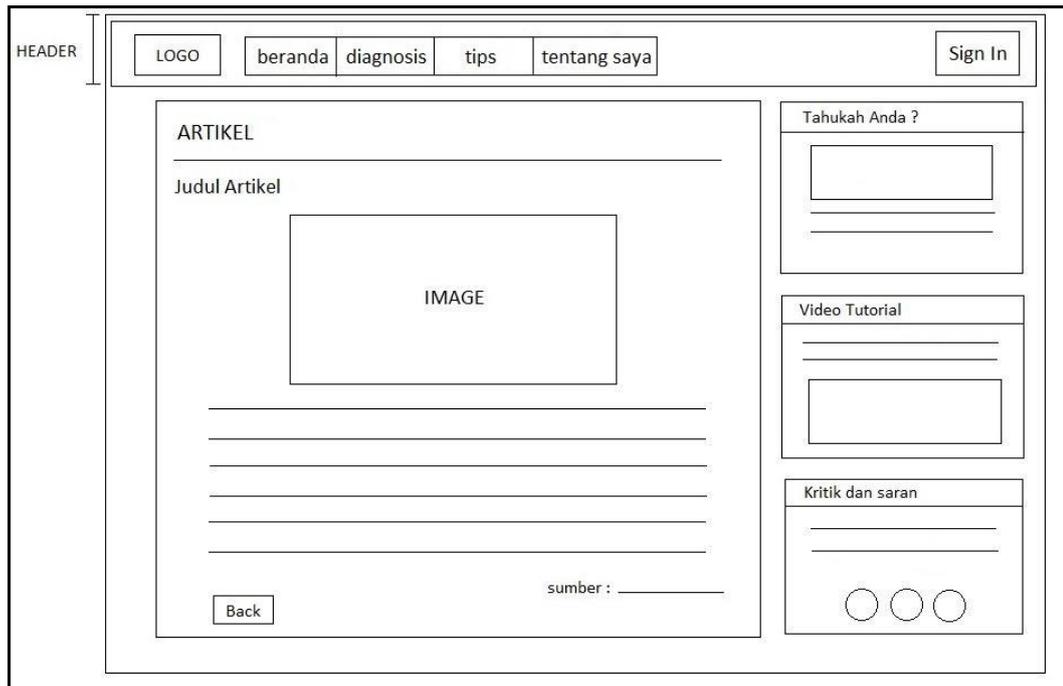
U
M
N



Gambar 3.15 Halaman Tips

Apabila salah satu *listview* ditekan maka halaman akan berganti ke halaman artikel. Halaman artikel menerima *id* dari *listview* yang ditekan sehingga konten yang tersedia pada halaman artikel akan bergantung sesuai dengan *listview* yang ditekan. Halaman artikel berhubungan langsung dengan *database* tips. Tampilan halaman artikel digambarkan pada Gambar 3.16.

UMMN

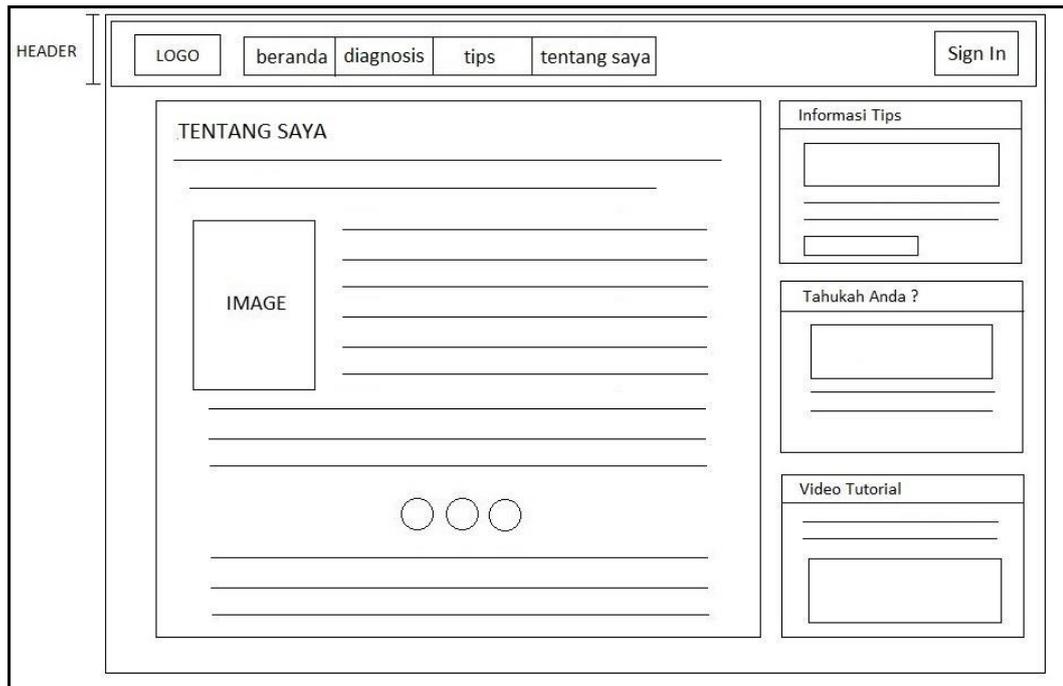


Gambar 3.16 Halaman Artikel

Halaman artikel berisikan informasi mengenai artikel yang ditekan melalui *listview* pada halaman tips. Konten utama dari artikel adalah *image*, berita, sumber, dan sebuah tombol untuk kembali ke halaman tips.

4. Halaman Tentang Saya

Halaman tentang saya merupakan bagian utama dari *frontend* website. Halaman ini berisikan sedikit informasi tentang perancang *website*. Halaman ini memiliki tiga bagian utama, yaitu *header*, konten utama, dan konten tambahan. Bagian *header* memiliki logo, menu, dan tombol *sign in*. Pada bagian konten utama terdapat *image*, penjelasan mengenai perancang *website*, dan tombol berbentuk lingkaran yang terhubung dengan media sosial. Bagian konten tambahan berisikan artikel tips yang dapat langsung terhubung ke halaman artikel, info sederhana, dan *video* tutorial. Tampilan halaman tentang saya digambarkan pada Gambar 3.17.



Gambar 3.17 Halaman Tentang Saya

5. Halaman Login

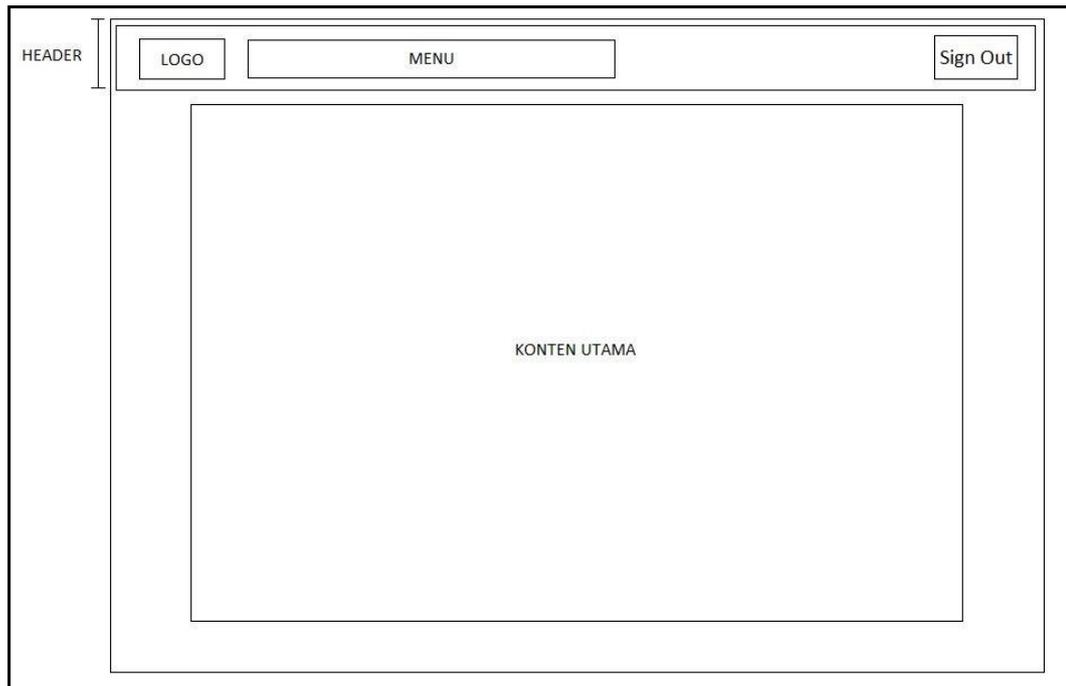
Halaman *login* memiliki tiga bagian utama, yaitu *header*, konten utama, dan konten tambahan. Bagian *header* memiliki logo, menu, dan tombol *sign in*. Pada bagian konten utama terdapat sebuah *alert message*, penjelasan mengenai cara *login*, dua buah *textbox* yang masing-masing digunakan untuk memasukkan *username* dan *password*, dan tombol masuk untuk menerima tampilan *admin*. Bagian konten tambahan berisikan artikel tips yang dapat langsung terhubung ke halaman artikel, info sederhana, dan kritik dan saran yang dapat dikirimkan melalui tombol lingkaran yang sudah terhubung langsung dengan media sosial. Tampilan halaman *login* digambarkan pada Gambar 3.18.

Gambar 3.18 Halaman *Login*

B. Rancangan Halaman Admin

Halaman *admin (backend)* ditujukan untuk *admin*. Untuk menerima tampilan *admin* dibutuhkan proses *login*.

Kerangka utama *backend* terdiri atas dua bagian utama, yaitu bagian *header* yang memiliki logo, menu, dan tombol *sign out*, dan bagian konten utama yang merupakan *container* dari konten utama dari halaman *backend*. Apabila layar diperkecil seukuran layar *smartphone* maka *header* hanya akan terdiri dari logo dan *toggle button* yang apabila ditekan akan memunculkan menu dan tombol *sign out*. Kerangka utama dari halaman *backend* dapat dilihat pada Gambar 3.19.

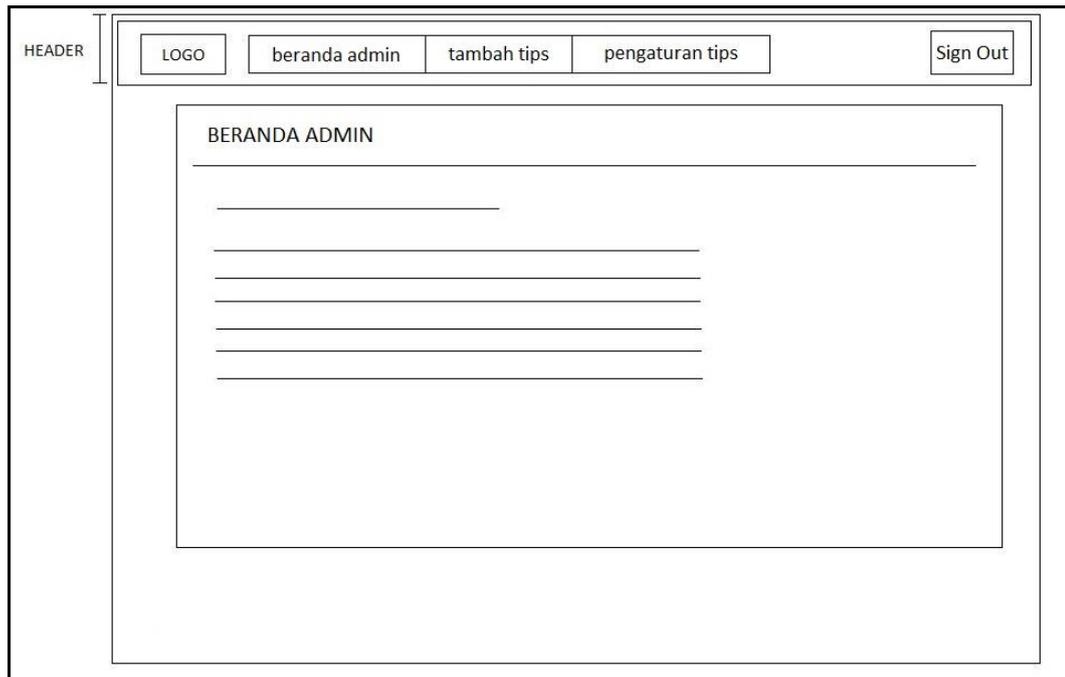


Gambar 3.19 Rancangan Halaman *Backend*

Halaman *backend website* memiliki tiga menu utama yaitu beranda admin, tambah tips, dan pengauran tips. Oleh karena itu, tampilan *backend* terdiri dari tiga halaman utama yang akan dijelaskan sebagai berikut.

1. Halaman Beranda Admin

Halaman beranda *admin* merupakan halaman utama dari *backend website*. Halaman ini memiliki dua bagian utama, yaitu *header* dan konten utama. Bagian *header* memiliki logo, menu, dan tombol *sign out*. Pada bagian konten utama terdapat penjelasan mengenai kegunaan dari halaman *backend*. Untuk lebih jelasnya, halaman beranda *admin* digambarkan pada Gambar 3.20.



Gambar 3.20 Halaman Beranda *Admin*

2. Halaman Tambah Tips

Halaman tambah tips merupakan bagian utama dari *backend website*. Halaman ini memiliki dua bagian utama, yaitu *header* dan konten utama. Bagian *header* memiliki logo, menu, dan tombol *sign out*. Pada bagian konten utama terdapat lima buah *textbox* yang digunakan untuk mengisi informasi mengenai tips yang ingin ditambahkan ke *database* serta dua tombol, yaitu tombol unggah yang digunakan untuk mengunggah gambar dan tombol simpan yang digunakan untuk menyimpan data. Halaman ini berhubungan langsung dengan *database* sehingga ketika tombol simpan ditekan maka data akan langsung masuk ke dalam *database*. Tampilan halaman tambah tips digambarkan pada Gambar 3.21.

HEADER

LOGO beranda admin tambah tips pengaturan tips Sign Out

TAMBAH TIPS

tambahkan data tips pada kolom di bawah ini kemudian tekan save untuk menyimpan

gambar unggah

judul

berita (teaser)

berita (full)

sumber

simpan

Gambar 3.21 Halaman Tambah Tips

3. Halaman Pengaturan Tips

Halaman pengaturan tips merupakan bagian utama dari *backend* website. Halaman ini memiliki dua bagian utama, yaitu *header* dan konten utama. Bagian *header* memiliki logo, menu, dan tombol *sign out*. Pada bagian konten utama terdapat sebuah penjelasan, dan sebuah tabel yang berisikan informasi mengenai tips yang ada dalam *database*. Terdapat dua tombol pada tabel, yaitu tombol ubah untuk berpindah ke halaman ubah tips dan tombol hapus untuk melakukan penghapusan tips. Halaman ini berhubungan langsung dengan *database* sehingga data tips pada tabel akan selalu sesuai dengan data dalam *database*. Tampilan halaman pengaturan tips digambarkan pada Gambar 3.22.

HEADER

LOGO beranda admin tambah tips pengaturan tips Sign Out

PENGATURAN TIPS

Tekan tombol 'ubah' untuk mengubah data atau tekan tombol 'hapus' untuk menghapus data

Tips yang ada					
#	judul	berita	sumber	edit	delete
1.	_____	_____	_____	ubah	hapus
2.	_____	_____	_____	ubah	hapus
3.	_____	_____	_____	ubah	hapus
4.	_____	_____	_____	ubah	hapus

Gambar 3.22 Halaman Pengaturan Tips

Apabila tombol ubah pada tabel ditekan maka halaman akan berpindah ke halaman ubah tips. Tampilan halaman ubah tips digambarkan pada Gambar 3.23.

HEADER

LOGO beranda admin tambah tips pengaturan tips Sign Out

UBAH TIPS

ubah data tips pada kolom di bawah ini kemudian tekan simpan untuk menyimpan

id _____

INFO : Unggah gambar baru untuk mengganti gambar yang sudah ada

gambar _____ unggah

judul _____

berita (teaser) _____

berita (full) _____

sumber _____

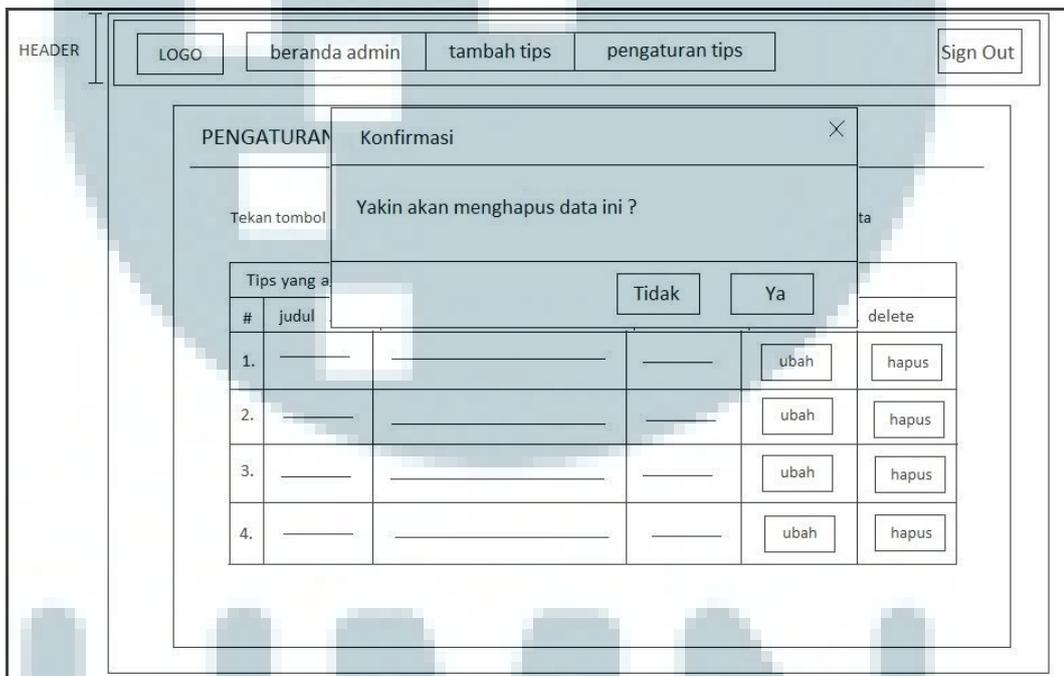
kembali simpan

Gambar 3.23 Halaman Ubah Tips

Halaman ubah tips terdiri dari penjelasan mengenai cara ubah, sebuah *alert*, enam buah *textbox*, dan dua buah tombol. Pada saat tombol ubah pada tabel yang

ada di halaman pengaturan tips ditekan maka *id* dari tips akan dikirimkan ke halaman ubah tips. Oleh karena itu, semua *textbox* pada halaman ini akan memiliki nilai *value* awal. *Textbox id* bersifat *readonly* sehingga tidak dapat diubah. Tombol kembali digunakan untuk kembali ke halaman pengaturan tips sedangkan tombol simpan digunakan untuk menyimpan data yang telah diubah ke dalam *database*.

Apabila tombol hapus pada tabel yang ada di halaman pengaturan tips ditekan maka akan muncul sebuah *modal* yang berfungsi sebagai pesan konfirmasi. Tampilan *modal* digambarkan pada Gambar 3.24.



Gambar 3.24 Tampilan Hapus *Modal*

Hapus *modal* berisikan pesan konfirmasi untuk melakukan penghapusan tips. *Modal* memiliki dua tombol, yaitu tidak dan ya. Apabila tombol tidak ditekan, maka *modal* akan menghilang. Apabila tombol ya ditekan, maka data tips pada *database* akan dihapus sesuai dengan *id* yang dikirimkan.