



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

METEDOLOGI PENELITIAN

3.1 Metodologi

Aplikasi sistem pakar yang berbasis web ini dibuat dengan menggunakan *neural network*. Langkah – langkah kerja penelitian adalah sebagai berikut.

1. Studi Literatur

Penulis melakukan studi terhadap referensi-referensi yang berhubungan dengan penyakit-penyakit yang telah ditentukan pada batasan masalah. Selain itu, penulis juga melakukan studi terhadap referensi-referensi yang berhubungan dengan *neural network* sebagai metode dalam pembangunan aplikasi sistem pakar. Referensi tersebut merupakan jurnal, buku, artikel dan hasil wawancara dengan pakar.

2. Observasi Penyakit-Penyakit yang Telah Ditentukan

Penulis melakukan observasi terhadap data-data yang berhubungan dengan penyakit-penyakit yang telah ditentukan pada batasan masalah untuk mendapatkan data-data mengenai gejala-gejala yang muncul dari penyakit, akibat yang ditimbulkan dari penyakit, penyebab timbulnya penyakit, dan cara mengobati penyakit itu sendiri.

3. Perancangan dan Pembangunan Aplikasi

Melakukan perancangan awal terhadap aplikasi yang akan dibangun, beserta dengan flow chart dari aplikasi yang dibuat, flow chat untuk algoritma yang digunakan dan struktur tabel yang digunakan.

4. Implementasi

Melakukan pembangunan aplikasi dengan mengimplemantasikan rancangan dan algoritma yang telah didefinisikan sebelumnya dengan menggunakan bahasa pemrograman yang telah ditentukan.

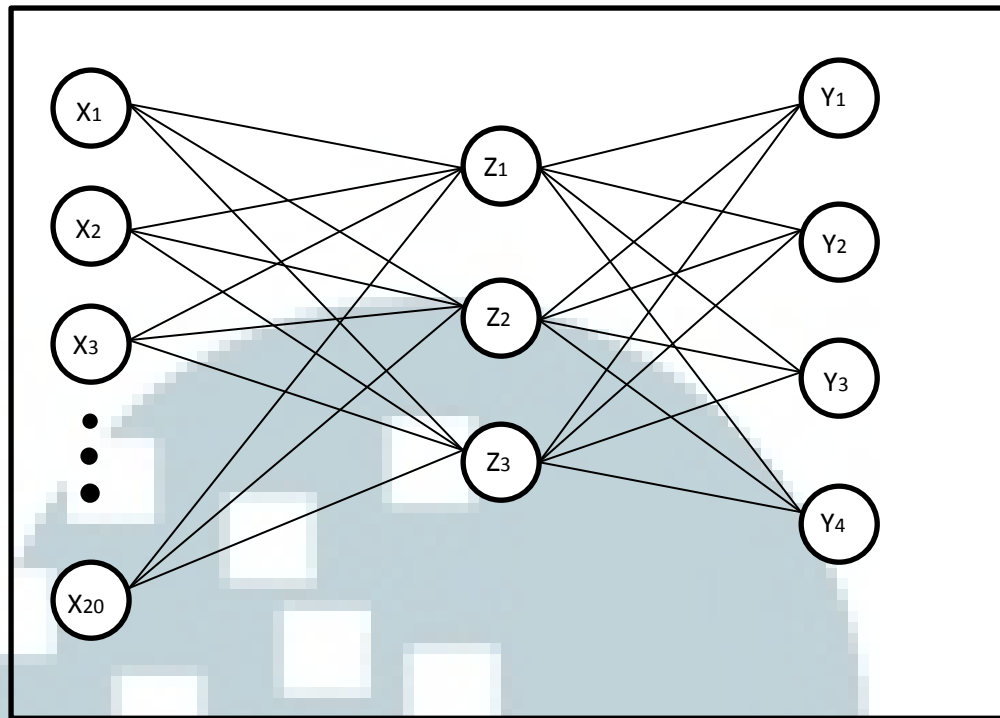
5. Uji Coba dan Evaluasi

Aplikasi pendeteksi penyakit pada alat pencernaan yang telah dibuat akan diuji coba. Percobaan meliputi seberapa cepat aplikasi akan mengeluarkan hasil. Percobaan juga meliputi seberapa akurat hasil yang dihasilkan oleh aplikasi tersebut, Hal ini bertujuan untuk mengetahui dan mencari bug yang muncul dari aplikasi tersebut serta untuk memperbaiki aplikasi tersebut.

3.2 Perancangan Sistem

3.2.1 Rancangan *Neural Network*

Pembangunan aplikasi menggunakan *Neural Network* dimana setelah melakukan diskusi dengan pakar, didapatkan input yang sebanyak 20 buah dan 4 buah output. Masing-masing input dan output merupakan angka biner antara 1 atau 0. Penggunaan biner ini, digunakan untuk memudahkan proses perhitungan, dimana output yang berupa daftar penyakit diumpamakan kedalam angka biner yang telah ditentukan penulis. Selain itu, input juga merupakan angka biner yang isinya bergantung akan gejala-gejala yang telah diberikan oleh pakar. Adapun rancangan awal dari neural network ini sendiri adalah:



Gambar 3.1 Rancangan awal neural network

Pada rancangan awal, penulis menggunakan 1 buah *hidden layer* dengan 3 buah *neuron*. Rancangan ini akan terus berubah seiring dengan proses pelatihan untuk mendapatkan hasil yang optimal. Berikut adalah tabel mengenai angka biner input yang digunakan pada arsitektur jaringan,

Tabel 3.1 Tabel input

Gejala	Ya	Tidak
Demam	1	0
Kontraksi perut (Perut kencang)	1	0
Tidak nafsu makan	1	0
Mual	1	0
Sulit buang air besar	1	0
Muncul benjolan saat mengejan	1	0

Gejala	Ya	Tidak
Rasa asam pada mulut	1	0
Reflux makanan	1	0
Sering mencejan saat buang air	1	0
Pendarahan pada anus	1	0
Terdapat sesuatu menonjol pada anus	1	0
Terlihat kuning pada beberapa bagian tubuh	1	0
Perasaan kembung	1	0
Seringnya telat makan	1	0
Buang air cair selama kurang dari 14 hari	1	0
Memiliki alergi makanan	1	0
Pernah mengalami bab cair akibat konsumsi makanan	1	0

Untuk gejala atau keluhan utama, diberikan input sebagai berikut,

Tabel 3.2 Gejala utama

Gejala	Input
Sakit pada lebih dari satu kuadran perut	001
Sakit pada perut bagian kanan atas	010
Sakit pada perut bagian kanan bawah	100
Sakit pada perut bagian kiri atas	101
Rasa terbakar sekitar kerongkongan hingga dada bagian bawah	011

Gejala	Input
Perasaan sakit pada anus	110
Terdapat benjolan sekitar perut bawah hingga selangkangan	111
Buang air besar cair (Mencret)	000

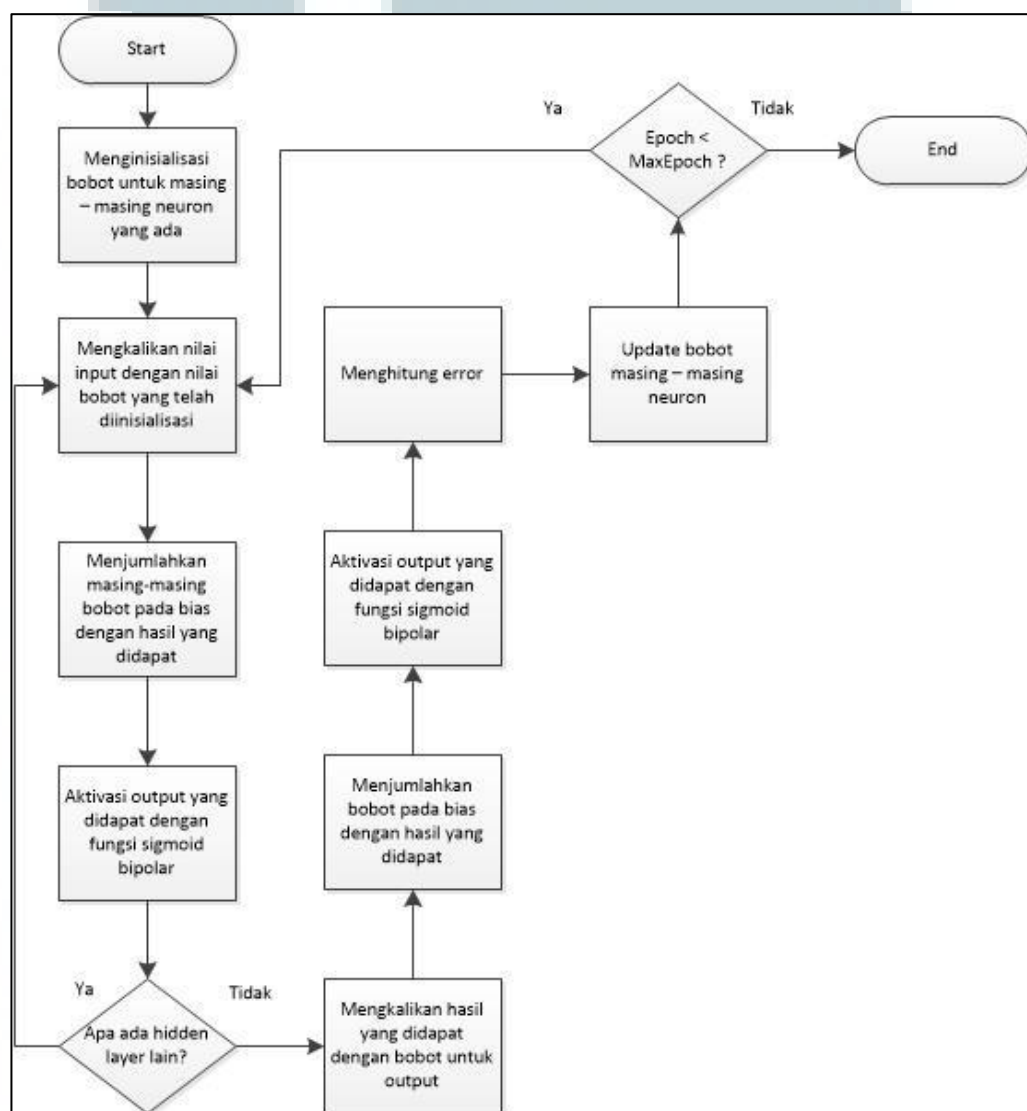
Dari gejala-gejala tersebut, data target yang digunakan adalah :

Tabel 3.3 Tabel hasil

Penyakit	Output
<i>Peritonitis</i>	0111
<i>Hernia</i>	1000
<i>GERD</i>	1010
<i>Hemmorhoids</i>	1001
<i>Appendicitis</i>	0001
<i>Cholelithiasis</i>	0010
<i>Gastritis</i>	0011
<i>Acute Diarrhea</i>	0100
<i>Choinic Diarrhea</i>	0101
<i>Intollerance Diarrea</i>	0110

3.2.2 Flowchart algoritma

Adapun perancangan aplikasi dibuat menggunakan bahasa pemrograman PHP dengan menggunakan algoritma *backpropagation* sebagai pendukung *neural network*. Algoritma *backpropagation* pada neural network sendiri terbagi menjadi dua tahap, yaitu *feed forward propagation* dan *feed backward propagation*.



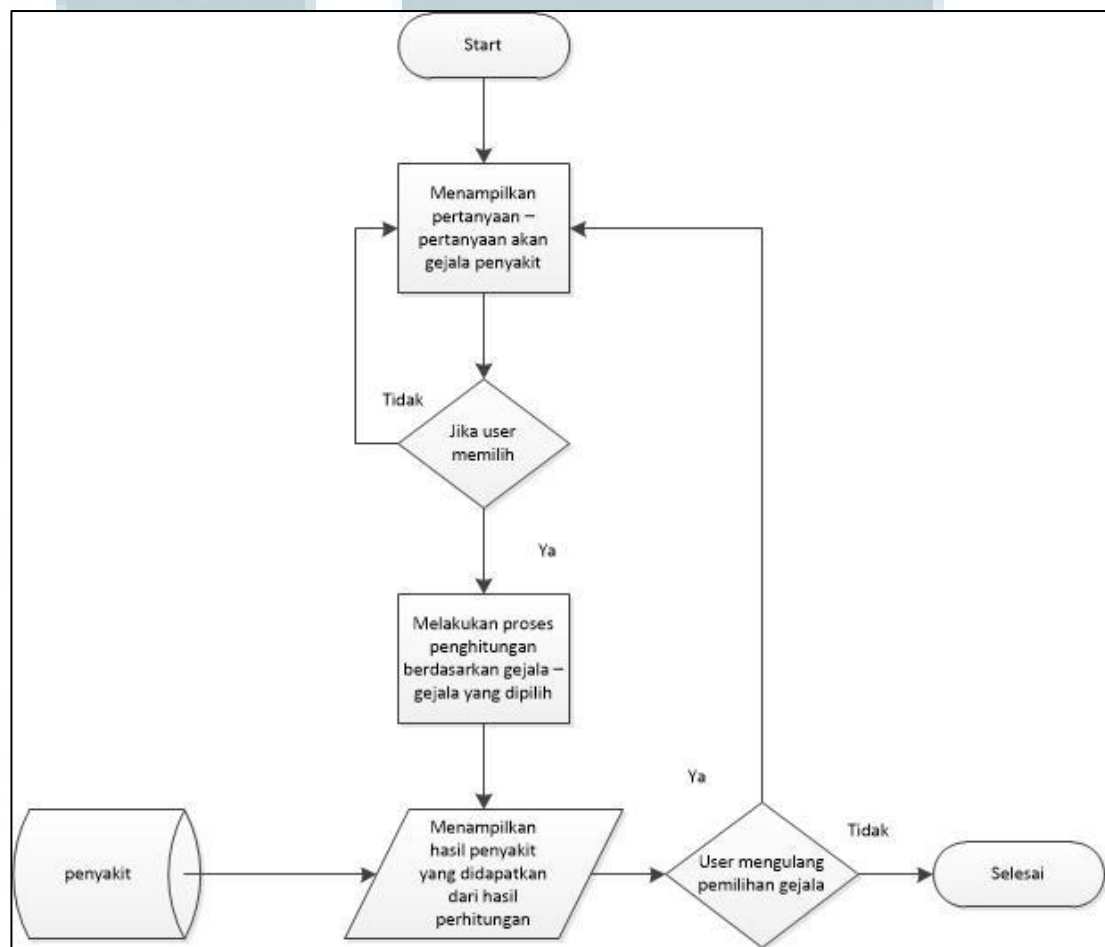
Gambar 3.2 Flowchart Algoritma

Pada algoritma *backpropagation neural network*, bobot awal yang diinisialisasi merupakan bobot dengan nilai random antara 0 – 1. Setiap input akan memiliki nilai bobot tersendiri untuk masing – masing neuron yang ada. Setelah bobot awal memiliki nilai, proses selanjutnya adalah mengkalikan masing – masing bobot dengan inputnya. Setelah proses selesai, maka proses selanjutnya adalah menjumlahkan bobot bias pada masing-masing *neuron* dengan hasil perkalian yang didapatkan. Lalu dilakukan *aktivasi* dengan menggunakan fungsi *sigmoid bipolar*. Apabila masih terdapat *hidden layer* lain, maka *output* yang didapat setelah hasil pengaktifan dengan fungsi *sigmoid* akan menjadi nilai *input* untuk *hidden layer* selanjutnya. Kemudian, mengulang proses diatas hingga *hidden layer* terakhir. Setelah selesai, maka kita dapat mendapatkan hasil *errornya* dengan mengurangi *output* yang didapat dengan target yang diinginkan. Proses inilah yang disebut dengan proses *feed forward propagation*. Setelah proses *feed forward propagation*, untuk mengurangi *error* yang didapat, maka dilakukan proses *feed backward propagation*. Proses ini merupakan proses dimana kita melakukan *update* nilai tiap-tiap bobot yang dimiliki. *Update* nilai bobot akan bergantung pada hasil *error* yang didapatkan. Setelah bobot diperbaharui, apabila proses belum mencapai iterasi (*Epoch*) maksimum, maka proses akan mengulang proses *feed forward propagation* dengan menggunakan nilai bobot yang telah diperbaharui, sedangkan jika telah mencapai nilai iterasi maksimum, maka proses akan berhenti dan berakhir. Proses diatas merupakan proses selama program belajar untuk mendapatkan nilai bobot yang optimal untuk outputnya. Apabila bobot optimal sudah didapatkan, maka bobot akan diuji

dengan input lain dengan hanya menggunakan proses *feed forward propagation* saja hingga mendapatkan nilai output. Nilai output tersebut akan dicocokkan sesuai dengan target yang diinginkan. Untuk mendapatkan jumlah *hidden layer* yang digunakan dan jumlah *neuron* yang digunakan, ditetapkan pada saat *learning program* untuk mendapatkan hasil *error* terrendah.

3.2.3 Flowchart program

Adapun *Flowchart* untuk aplikasi yang dibuat adalah,

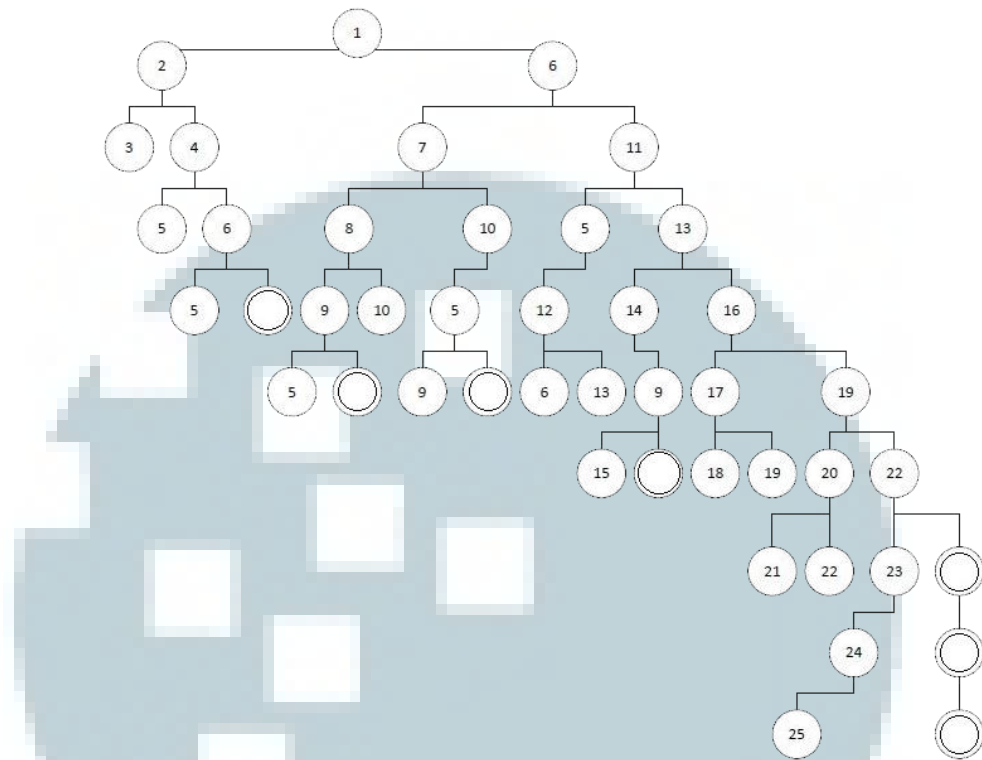


Gambar 3.3 Flowchart Program

Pertama kali, aplikasi akan menampilkan pertanyaan-pertanyaan yang bersangkutan dengan gejala-gejala penyakit yang memungkinkan. Apabila user memilih gejala, maka program akan terus menampilkan pertanyaan hingga semua pertanyaan yang diberikan telah terjawab. Setelah itu, gejala-gejala yang telah dipilih akan menjadi suatu nilai input untuk proses pencarian penyakit berdasarkan algoritma yang telah dijelaskan diatas. Setelah proses perhitungan selesai, aplikasi akan menampilkan data penyakit yang sesuai dengan gejala-gejala yang dipilih. Kemudian, apabila user ingin melakukan pemilihan gejala ulang, maka program akan mulai menampilkan pertanyaan-pertanyaan kembali, sedangkan apabila user tidak ingin melakukan pemilihan gejala ulang, maka program akan berakhir.

3.2.4 *Pohon keputusan*

Aplikasi yang dibuat, menggunakan beberapa pertanyaan yang bersangkutan dengan gejala-gejala penyakit untuk mendapatkan input yang akan digunakan dalam perhitungan program. Pertanyaan-pertanyaan yang diberikan dapat dibentuk dalam sebuah bentuk *tree* dimana pertanyaan yang diberikan hanya memiliki jawaban antara ya atau tidak. Apabila user mengisikan ya, maka *tree* akan berjalan kearah kiri, jika tidak maka *tree* bergerak ke arah kanan. Berikut adalah bentuk dari *tree* yang dibuat :



Gambar 3.0.4 Tree pertanyaan

UMN

Pohon keputusan yang dibuat adalah pohon keputusan untuk pertanyaan-pertanyaan yang di perumpamakan dalam angka. Adapun pertanyaan-pertanyaan tersebut diatas adalah sebagai berikut.

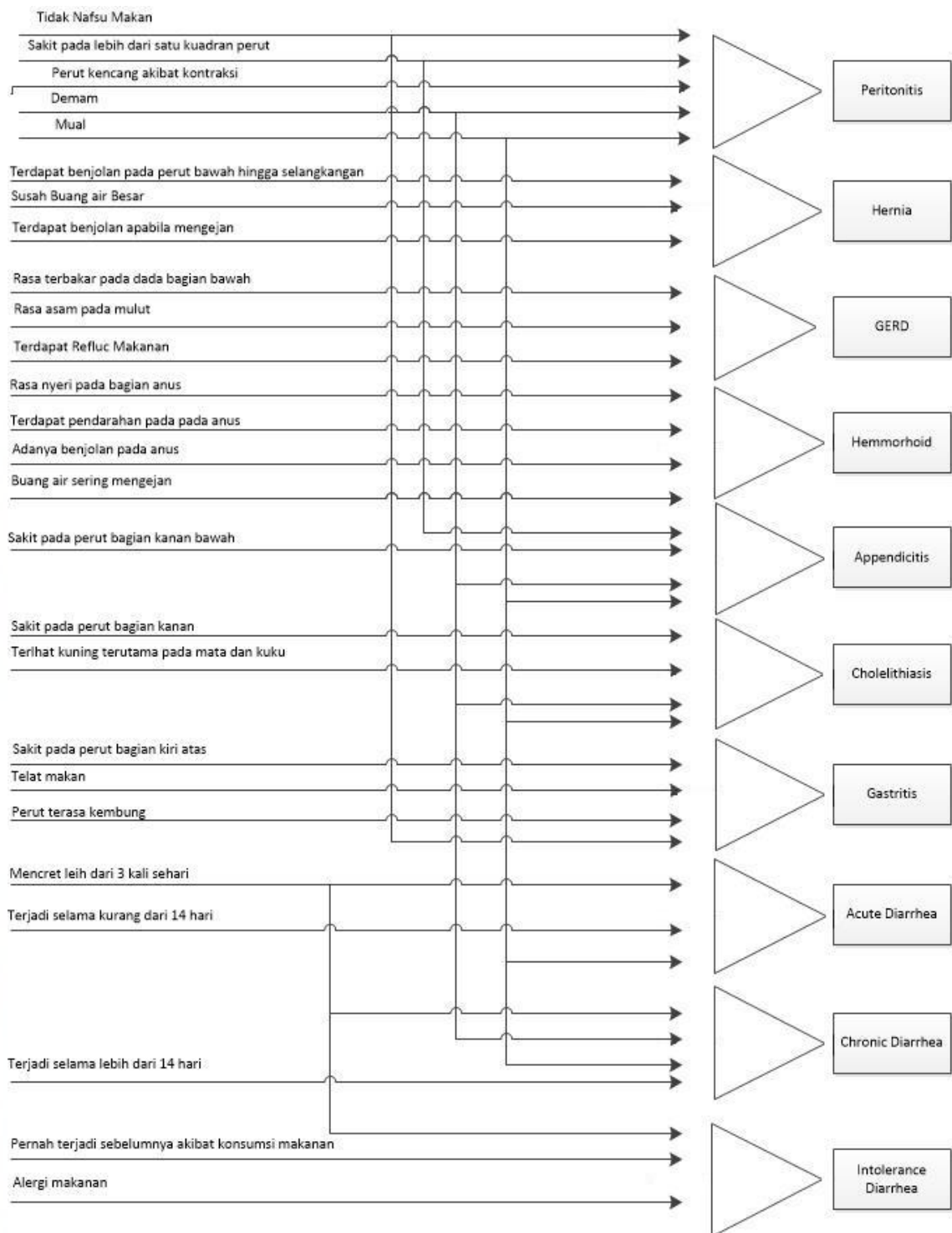
1. Apakah terdapat mencret yang terjadi lebih dari tiga kali sehari?
2. Apakah penyakit ini pernah berulang akibat konsumsi suatu makanan yang sama?
3. Apakah anda memiliki alergi pada suatu makanan?
4. Apakah mencret telah terjadi selama kurang dari 14 hari?
5. Apakah terdapat perasaan mual?
6. Apa ada demam?
7. Apakah terdapat rasa sakit pada lebih dari satu kuadran perut?
8. Apakah perut anda terasa kencang akibat adanya kontraksi pada perut?
9. Apakah anda merasa tidak nafsu makan?
10. Apa terdapat rasa sakit pada perut bagian kiri bawah?
11. Apa terdapat rasa sakit pada perut bagian kanan atas?
12. Apakah terdapat bagian yang terlihat kuning (terutama mata dan kuku)?
13. Apakah terdapat rasa sakit pada perut bagian kiri atas?
14. Apakah anda sering telat makan?
15. Apakah perut anda terasa agak kembung?
16. Apakah anda merasakan rasa terbakar sekitar kerongkongan atau dada bagian bawah?
17. Apa mulut anda terasa asam?

18. Apakah terjadi *reflux* makanan(peristiwa keluarnya kembali makanan yang dikonsumsi)?
19. Apa terdapat benjolan sekitar perut bawah hingga selangkangan?
20. Apabila anda mengejan, apakah muncul suatu benjolan sekitar perut bawah hingga selangkangan?
21. Apakah anda sulit buang air besar?
22. Apakah terdapat rasanyeri sekitar anus?
23. Apa terdapat pendarahan pada bagian anus(darah berwarna merah cerah)?
24. Apakah ada yang menonjol sekitar area anus?
25. Apakah anda sering mengejan saat buang air besar?

Setelah pertanyaan-pertanyaan terjawab oleh user, maka aplikasi akan melakukan kalkulasi dan mendapatkan nilai output, untuk hubungan ketergantungan antara faktor-faktor yang mempengaruhi dalam penentuan penyakit akan dibahas dalam dependency diagram.

3.2.5 Depedency diagram

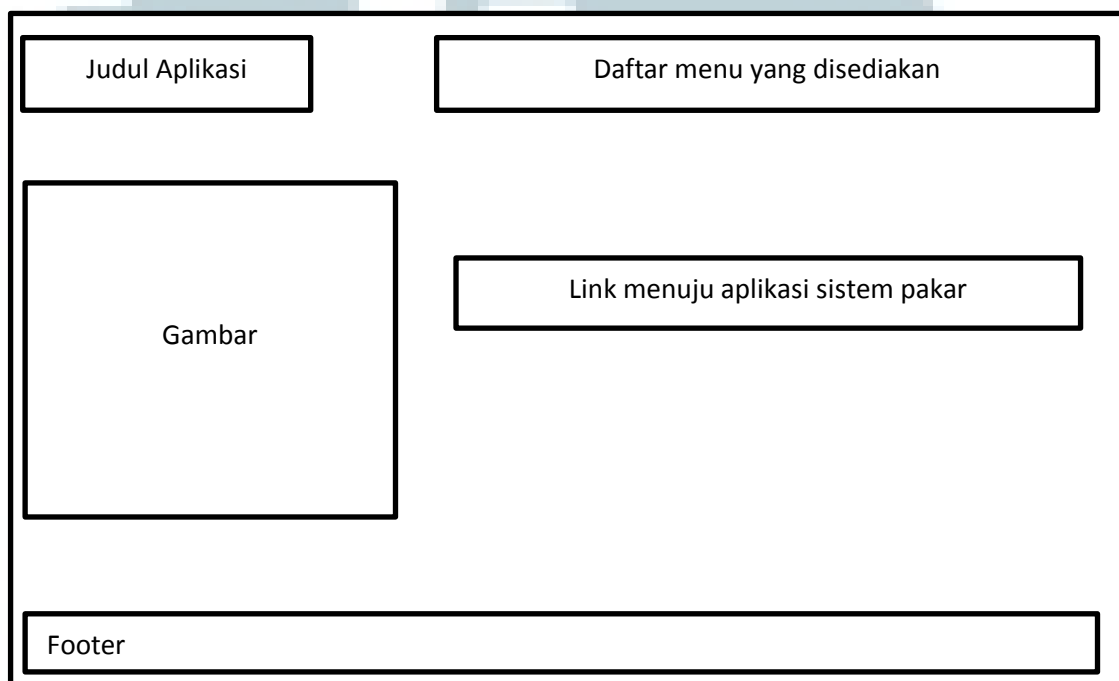
Dependency diagram dibuat untuk menggambarkan faktor-faktor yang mempengaruhi dalam penentuan penyakit. berikut adalah *dependency* diagram pada aplikasi sistem pakar ini.



Gambar 3.5 Depedency Diagram

3.3 Sketsa Layar

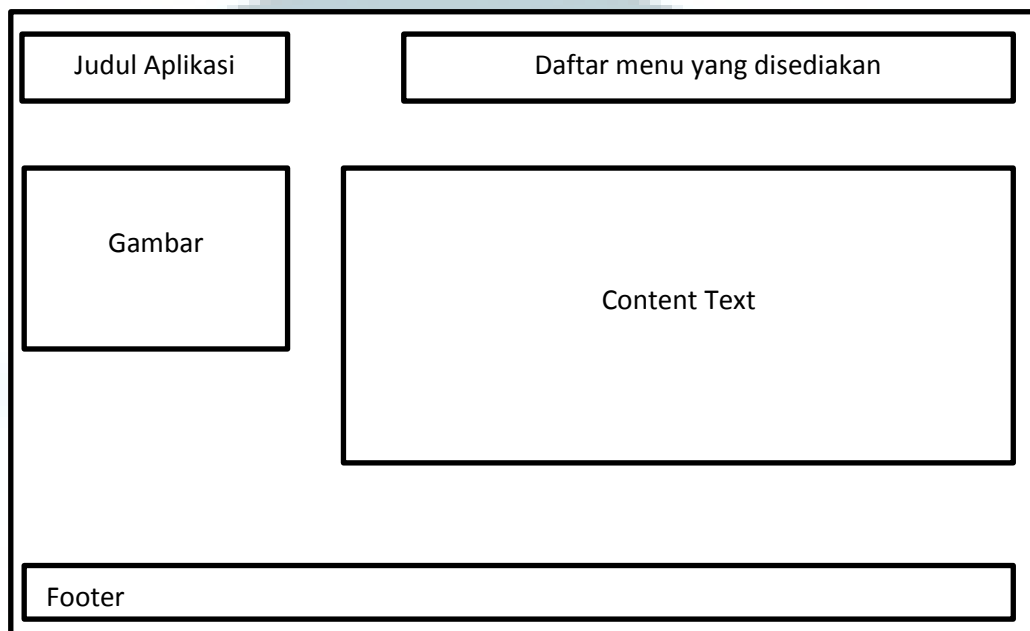
Aplikasi dibuat berbasis web dengan lima halaman utama yaitu, halaman utama (home), halaman about (tentang web), halaman disease (daftar penyakit yang ada), halaman expert system (halaman untuk aplikasi), dan halaman contact (halaman mengenai pembuat aplikasi). Untuk setiap halaman, disediakan navigasi berupa menu-menu untuk menuju halaman-halaman yang disediakan, adapun sketsa untuk halaman utama adalah :



Gambar 3.6 Sketsa halaman utama

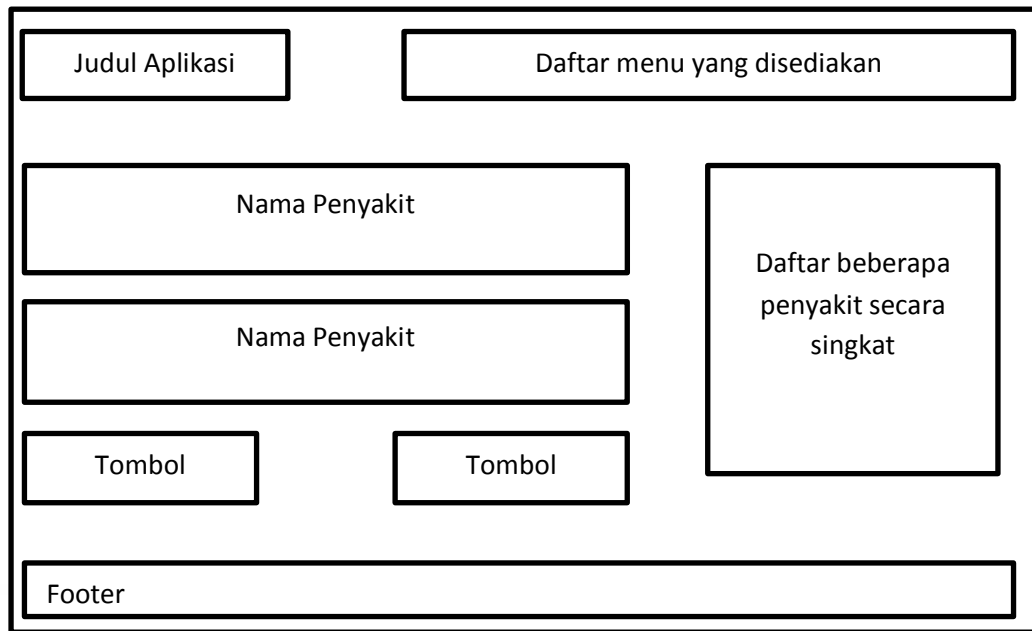
Gambar diatas merupakan sketsa untuk halaman utama aplikasi pada web. Pada daftar menu yang disediakan, user dapat mengakses halaman-halaman yang telah disediakan, yaitu halaman *about*, halaman *contact*, halaman daftar penyakit dan halaman untuk aplikasi sistem pakar. Menu ini disediakan untuk setiap

halaman untuk mempermudah user berpindah halaman. Untuk halaman about, diberikan sketsa sebagai berikut,



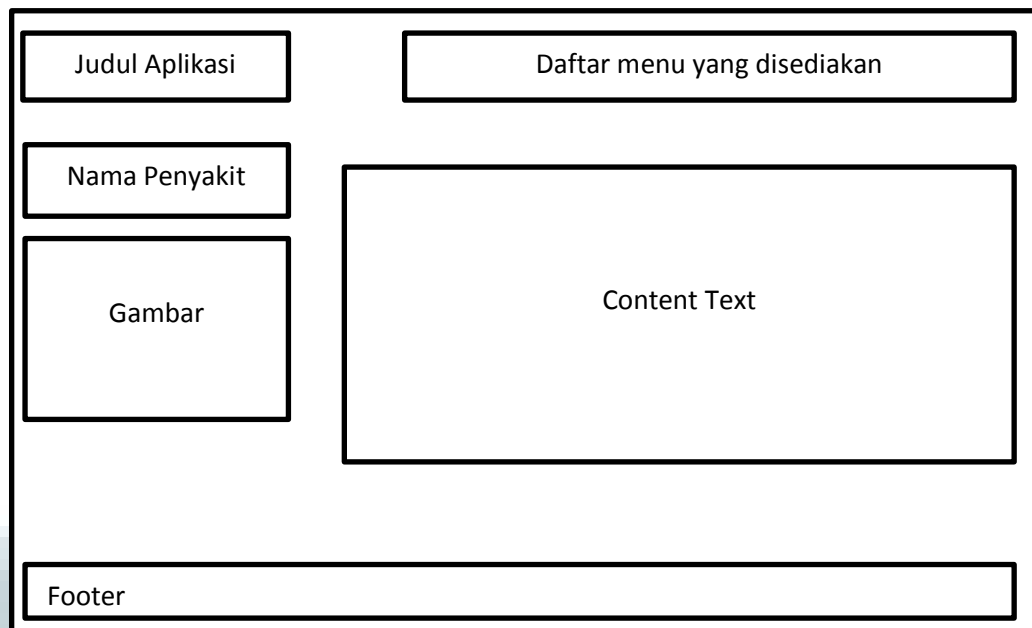
Gambar 3.7 Sketsa halama About

Halaman ini berisikan data-data mengenai web dan aplikasi yang dibuat oleh penulis, berisikan gambar halaman utama *web*, dan content text yang berisikan informasi-informasi akan web yang dibuat. Untuk halaman *disease* (halaman daftar penyakit), diberikan sketsa halaman sebagai berikut,



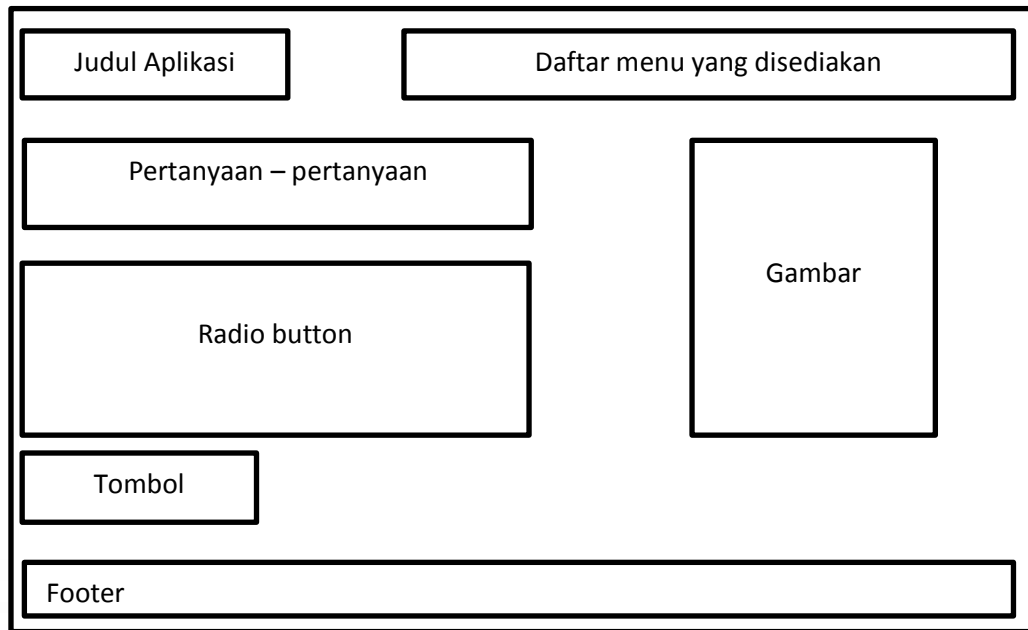
Gambar 3.8 Sketsa halaman Disease

Pada halaman ini, user dapat memilih nama penyakit untuk melihat detail dari penyakit itu sendiri. Halaman ini menampilkan dua penyakit setiap halamannya. Untuk melihat daftar penyakit selanjutnya dapat mengklik tombol yang disediakan dan halaman akan menampilkan penyakit lainnya. Pada kanan content halaman, disediakan daftar penyakit yang sering dibuka oleh user, di urutkan berdasarkan berapa sering user membuka halaman detail penyakit itu dari halaman daftar penyakit maupun halaman pada aplikasi sistem pakar. Apabila user mengklik nama penyakit yang disediakan, maka halaman akan berpindah pada halaman detail penyakit. Adapun sketsa halaman tersebut adalah :



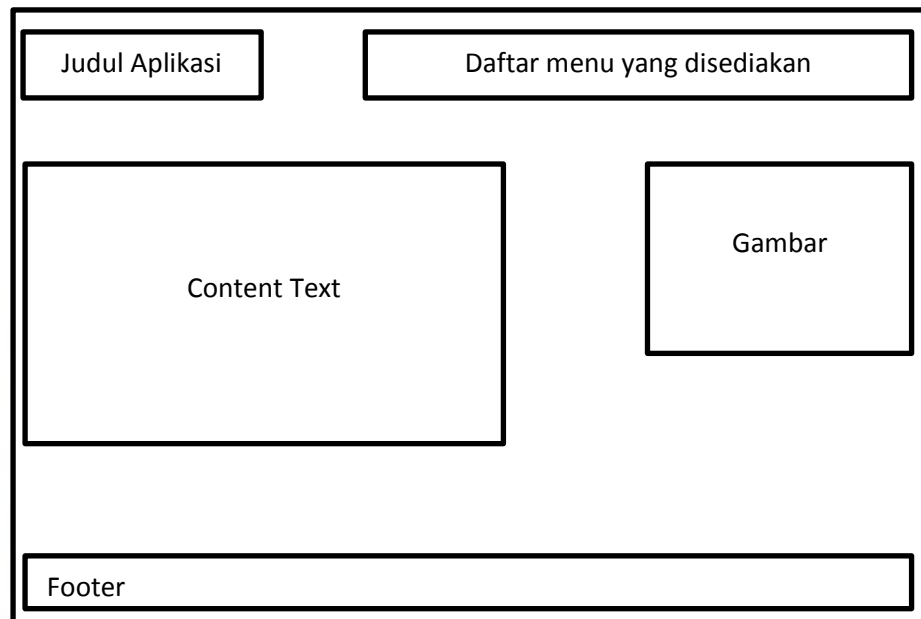
Gambar 3.9 Sketsa halaman detil penyakit

Halaman ini berisikan informasi-informasi yang bersangkutan dengan penyakit yang dipilih oleh *user*. Informasi ini berkait dengan deskripsi penyakit dan gejala-gejala yang bersangkutan pada penyakit itu sendiri. Informasi itu diletakan pada bagian *content text* pada halaman *web*. Untuk halaman aplikasi sistem pakar, merupakan halaman yang berisikan pertanyaan-pertanyaan yang besangkutan dengan gejala-gejala yang nantinya akan menjadi suatu input yang digunakan dalam penghitungan dan akan mengeluarkan hasil. Untuk halaman ini, diberikan sketsa sebagai berikut,



Gambar 3.10 Sketsa halaman expert system

Halaman ini akan menampilkan pertanyaan-pertanyaan yang akan berkaitan dengan pemilihan gejala-gejala. User akan menjawab pertanyaan dengan memilih salah satu pilihan yang diberikan. Setelah menjawab semua pertanyaan, maka aplikasi akan melakukan penghitungan dan menampilkan penyakit sesuai dengan hasil perhitungan. Halaman yang ditampilkan adalah halaman detil penyakit yang telah dibahas sebelumnya. Proses penghitungan input yang didapatkan tidak ditampilkan pada halaman. Halaman hanya menampilkan halaman hasil dari perhitungan, yang selanjutnya akan menuju ke halaman yang merupakan hasil dari perhitungan. Untuk halaman *contact* diberikan sketsa sebagai berikut,



Gambar 3.11 Sketsa halaman contact

Halaman ini berisikan informasi-informasi mengenai penulis yang selaku pembuat web beserta keterangan-keterangan lainnya.

3.4 Struktur tabel

Database yang digunakan pada pembangunan aplikasi adalah MySQL Server 1.81 dengan nama database “sistempakar” dan menggunakan 2 buah tabel.

Berikut adalah daftar tabel yang digunakan :

Nama Tabel : penyakit

Fungsi : menyimpan data mengenai penyakit-penyakit yang dibahas oleh penulis.

Primary Key : kode_penyakit

Tabel 3.4 Tabel penyakit

No	Nama Field	Tipe data	Panjang	Keterangan
1	Id	Int		Id unik yang berbeda satu sama lain dengan data lain
2	kode_penyakit	Varchar	5	<i>Primary key</i> untuk tabel penyakit
3	nama_penyakit	varchar	50	Nama penyakit yang ada
4	deskripsi	MediumText		Deskripsi dari penyakit
5	gambar	Varchar	50	Lokasi gambar untuk penyakit
6	gambardetail	Varchar	50	Lokasi gambar untuk penyakit
7	gambaraside	Varchar	50	Lokasi gambar untuk penyakit
8	Nilai_output	Varchar	4	Nilai output berupa biner
9	view	Int		Jumlah penyakit dibaca

Nama Tabel : gejala

Fungsi : menyimpan gejala – gejala berdasarkan kode penyakit

Foreign Key : kode_penyakit

Tabel 3.5 Tabel gejala

No	Nama Field	Tipe data	Panjang	Keterangan
1	Id	Int		Id unik yang berbeda satu sama lain dengan data lain
2	kode_penyakit	Varchar	5	<i>foreign key</i> untuk tabel gejala ke tabel penyakit
3	gejala	varchar	100	Nama gejala yang ada

Nama Tabel : weight

Fungsi : menyimpan bobot untuk input ke *hidden layer*.

Primary Key :-

Foreign Key :-

Tabel 3.6 Tabel bobot untuk input

No	Nama Field	Tipe data	Panjang	Keterangan
1	ID	Int		Auto Increment
2	Layer	Int		Informasi <i>layer</i>
3	neuron	Int		Nomor <i>Neuron</i>
4	bobot	Double		<i>foreign key</i> untuk tabel gejala ke tabel penyakit

Nama Tabel : bias

Fungsi : menyimpan nilai bobot bias yang digunakan.

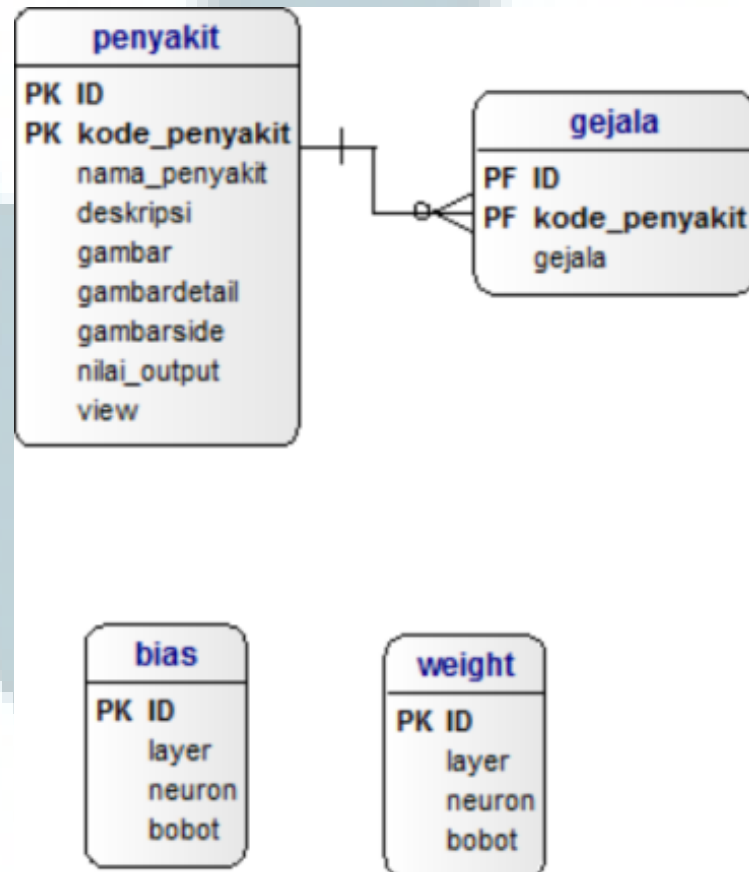
Primary Key :-

Foreign Key :-

Tabel 3.7 Tabel bobot bias

No	Nama Field	Tipe data	Panjang	Keterangan
1	ID	Int		Auto Increment
2	Layer	Int		Informasi layer
3	Neuron	Int		Nomor <i>neuron</i>
4	bobot	Double		Bobot

Adapun hubungan antar tabel yang dibuat adalah sebagai berikut.



Gambar 3.12 Hubungan antar tabel

UMMN