

Sistem Pakar Mendiagnosis Penyakit Jantung dengan Metode *Fuzzy Set*

Derist Touriano¹

Teknik Informatika, STIKOM Dinamika Bangsa
Jambi, Indonesia
Amuntai9@ovi.com

Erick Fernando²

Sistem Komputer, STIKOM Dinamika Bangsa
Jambi, Indonesia
Erick.fernando_88@yahoo.com

Pandapotan Siagian³

Teknik Informatika, STIKOM Dinamika Bangsa
Jambi, Indonesia
siagian.p@gmail.com

Hetty Rohayani. AH⁴

Sistem Komputer, STIKOM Dinamika Bangsa
Jambi, Indonesia
hetty_mna@yahoo.com

Abstract—Penyakit jantung merupakan penyakit yang menyebabkan tingkat kematian yang sangat tinggi. Ini yang disebabkan kurangnya pengetahuan masyarakat dan seorang pakar atau dokter ahli yang memiliki kelemahan dalam pelayanan seperti jam kerja terbatas dan menunggu antrian dalam melakukan pelayanan. Dengan demikian dibutuhkan untuk membuat komputer yang dapat bekerja seperti pakar atau dokter yang di sebut sebuah Sistem pakar (expert system) yang dibangun dalam bentuk website dengan menggunakan metode logika fuzzy. Logika Fuzzy merupakan peningkatan terhadap logika Boolean yang telah mengenalkan konsep kebenaran sebagian. Logika fuzzy dapat menggantikan kebenaran boolean dengan tingkat kebenaran yang lebih baik. Dengan Sistem pakar dibangun pada sebuah website yang dapat diakses oleh publik dimana saja dan kapanpun juga sehingga dapat membantu memberikan informasi yang sangat dibutuhkan secara cepat, akurat dan jelas yang layaknya berkonsultasi kepada seorang pakar atau pun seorang dokter ahlinya sehingga dapat mengurangi tingkat kematian yang terjadi.

Keywords—Sistem pakar, Fuzzy logic, penyakit jantung

I. PENDAHULUAN

Penyakit jantung merupakan penyakit yang menyebabkan tingkat kematian yang sangat tinggi. Ini yang disebabkan kurangnya pengetahuan masyarakat tentang penyakit ini. Dengan demikian masyarakat lebih mempercayakannya kepada seorang pakar atau dokter ahli tetapi walaupun adanya pakar atau dokter ahli, terkadang memiliki kelemahan seperti jam kerja terbatas dan menunggu antrian dalam melakukan pelayanan. Dengan demikian dibutuhkan untuk membuat komputer yang dapat bekerja seperti pakar atau dokter yang di sebut sebuah Sistem pakar (expert system). Penggunaan sistem pakar dapat membantu manusia untuk mencari jawaban dan solusi. Sistem pakar merupakan sebuah kecerdasan buatan yang terdapat dalam sebuah perangkat lunak yang dibangun dengan kemampuan mendekati seorang pakar (manusia) yang memiliki pengetahuan tinggi dalam sebuah bidang tertentu yang diharapkan dapat membantu memecahkan sebuah masalah. Sistem pakar memiliki banyak metode yang dapat digunakan salah satu ada logika fuzzy. Logika Fuzzy adalah peningkatan dari logika Boolean yang mengenalkan konsep kebenaran sebagian. Di mana logika klasik menyatakan bahwa

segala hal dapat diekspresikan dalam istilah biner yaitu 0 atau 1, logika *fuzzy* menggantikan kebenaran boolean dengan tingkat kebenaran. Sistem pakar ini dibangun pada sebuah website yang dapat diakses oleh publik secara realtime, dimana saja dan kapanpun juga sehingga dapat membantu memberikan informasi yang sangat dibutuhkan secara cepat, akurat dan jelas yang layaknya berkonsultasi kepada seorang pakar atau pun seorang dokter ahlinya.

II. LANDASAN TEORI

A. Sistem Pakar

Sistem pakar melakukan penggabungan dari pengetahuan yang ada dan penelusuran data untuk memecahkan masalah yang secara normal memerlukan keahlian dari seorang manusia. Tanpa adanya keahlian manusia sistem pakar tidak dapat berjalan karena ia memberikan keputusan seperti manusia. Dari hal ini penulis memberikan beberapa definisi dari sistem pakar berikut. Sistem pakar (*Expert System*) adalah sistem yang berusaha mengadopsi pengetahuan manusia ke komputer, agar komputer dapat menyelesaikan masalah seperti yang biasa dilakukan para ahli. Sistem pakar merupakan sebuah sistem yang mengadopsi pengetahuan yang dimiliki oleh pakar (manusia) ke dalam komputer, agar komputer dapat menyelesaikan sebuah masalah layaknya seorang ahli menyelesaikan masalah.[1]

B. Logika Fuzzy

Logika *Fuzzy* adalah sebuah logika yang merupakan peningkatan dari logika Boolean yang mengenalkan konsep kebenaran sebagian. Di mana logika klasik menyatakan bahwa segala hal dapat diekspresikan dalam istilah biner yaitu 0 atau 1, logika *fuzzy* menggantikan kebenaran boolean dengan tingkat kebenaran. Sebagaimana yang dikemukakan Lotfi A. Zadeh : “Logika *fuzzy* merupakan generalisasi dari logika klasik yang hanya memiliki dua nilai keanggotaan antara 0 dan 1.”[9]

Logika fuzzy memiliki nilai kemungkinan sama dengan logika probabilitas akan tetapi dalam konsep berbeda karena logika fuzzy sesuai dengan derajat kebenaran sedangkan logika probabilitas sesuai dengan nilai kemungkinan. Seperti yang dikemukakan Sri Kusuma Dewi : “karena ketidaksamaan

pada interpretasi-interpretasi teori probabilitas. Logika *fuzzy* sesuai dengan derajat kebenaran, sementara logika probabilistik sesuai dengan kemungkinan.”

Alasan mengapa menggunakan logika *fuzzy* antara lain:

- Konsep logika *fuzzy* mudah dimengerti.
- Logika *fuzzy* sangat fleksibel
- Logika *fuzzy* memiliki toleransi terhadap data-data yang tidak tepat.
- Logika *fuzzy* mampu memodelkan fungsi-fungsi non linear yang sangat kompleks.
- Logika *fuzzy* dapat membangun dan mengaplikasikan pengalaman-penagalaman para pakar secara langsung tanpa harus melalui proses pelatihan.
- Logika *fuzzy* dapat bekerja sama dengan teknik-teknik kendali secara konvensional.
- Logika *fuzzy* didasarkan pada bahasa alami.[3]

C. Fuzzy Set

Logika *fuzzy* merupakan generalisasi dari logika klasik yang hanya memiliki dua nilai keanggotaan, yaitu 0 dan 1. Dalam logika *fuzzy* nilai kebenaran suatu pernyataan berkisar dari sepenuhnya benar sampai sepenuhnya salah. Dengan teori *fuzzy set*, suatu objek dapat menjadi anggota dari banyak himpunan dengan derajat keanggotaan yang berbeda dalam masing-masing himpunan. Konsep ini berbeda dengan teori himpunan klasik (*crisp*). Teori himpunan klasik tergantung pada logika dua-nilai (*two-valued logic*) untuk menentukan apakah sebuah objek merupakan suatu anggota himpunan atau bukan [2]

Knowledge-based fuzzy set adalah merupakan suatu logika *fuzzy* yang digunakan untuk menyatakan suatu ketidakpastian dalam menentukan keanggotaan dari suatu elemen terhadap suatu *set* dengan memberikan derajat keanggotaan (*membership degree*) antara 0 sampai dengan 1 yang diberikan kepada beberapa orang (*knowledge*). Definisi *knowledge-based fuzzy set* adalah sebagai berikut: Misal $U = \{u_1, \dots, u_n\}$ sebagai *set of element* dan $K = \{k_1, \dots, k_n\}$ sebagai *set of knowledge*, kemudian suatu *fuzzy set* A , $k_1(A)$ didefinisikan sebagai sebuah *fuzzy set* berdasarkan *knowledge* k_1 terhadap *universal set* U dengan suatu *mapping* dari U ke dalam interval yang tertutup $[0,1]$. [2]

$$\mu_{k_1}(A) : U \rightarrow [0,1]$$

III. ANALISA DAN PERANCANGAN SISTEM

A. Analisa Kebutuhan variable

Variabel kebutuhan yang digunakan pada sistem pakar ini yaitu data penyakit jantung dan data gejala penyakit jantung. Akan dijabarkan pada tabel 3.1 dan 3.2

Tabel 3.1 Kode Variabel Gejala Penyakit Jantung

Kode Variabel Gejala	Nama Gejala
BB-1	Batuk
BB-2	Buang air kecil
DD-1	Demam menggigil

DD-2	Dehidrasi
GG-1	Gelisah
KK-1	Kesulitan menelan
KK-2	Kesemutan
LL-1	Lemah
LL-2	Lelah
MM-1	Mengalami dingin pada anggota tubuh
NN-1	Nyeri dada
NN-2	Nyeri pada tangan, punggung, dan perut
NN-3	Nafsu tidur
NN-4	Nafas cepat
PP-1	Pusing
PP-2	Penurunan kesadaran
PP-3	Pembengkakan pada kaki
PP-4	Penurunan berat badan
PP-5	Pingsan
PP-6	Pendarahan
PP-7	Peningkatan nadi
SS-1	Stres
SS-2	Sesak nafas
SS-3	Sakit kepala
SS-4	Suara serak
WW-1	Warna kulit pucat

Tabel 3.2 Kode Variabel Penyakit Jantung

Kode Penyakit Jantung	Nama Penyakit
A-1	Atherosclerosis
A-2	Aortic Aneurysm
A-3	Arterial Embolism
C-1	Cardiorespiratory Arrest
H-1	Hypovolemic Shock
J-1	Jantung Koroner
S-1	Septick Shock

B. Perancangan Mesin Inferensi

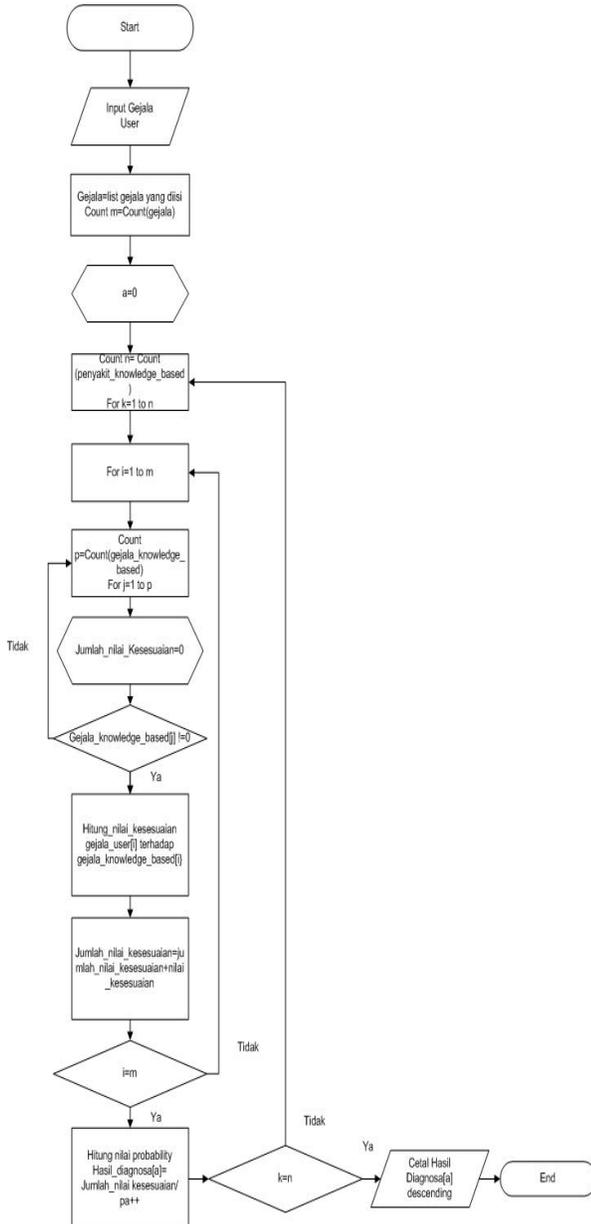
Pada perancangan sistem pakar untuk mendiagnosis suatu penyakit dilakukan dengan menggunakan mesin inferensi. Mesin *inferensi forward chaining* digunakan dalam sistem ini untuk mendiagnosis suatu penyakit setelah menerima gejala-gejala yang *diinput* oleh *user*. Setiap jawaban yang akan diberikan oleh seorang *user* akan dicari nilai kesesuaian/kesamaanya dengan nilai gejala suatu penyakit tertentu yang ada di dalam *knowledge-based* tersebut, sehingga diperoleh nilai kesesuaian berdasarkan frekuensi dan intensitas untuk masing-masing data gejala yang telah *diinput* pada penyakit tertentu.

Setelah mendapatkan nilai kesesuaiannya, kemudian dilakukan pengelompokan nilai kesesuaian untuk setiap penyakit lalu menjumlahkannya. Setelah mendapatkan jumlah nilai kesesuaian untuk setiap penyakit, selanjutnya mencari nilai *probability* untuk setiap penyakit dengan cara membandingkan jumlah nilai kesesuaian setiap penyakit dengan banyaknya gejala yang dimiliki oleh penyakit tersebut di *knowledge-based*.

Setelah mendapatkan nilai kesesuaiannya, kemudian dilakukan pengelompokan nilai kesesuaian untuk setiap penyakit lalu menjumlahkannya. Setelah mendapatkan jumlah nilai kesesuaian untuk setiap penyakit, selanjutnya mencari nilai *probability* untuk setiap penyakit dengan cara

membandingkan jumlah nilai kesesuaian setiap penyakit dengan banyaknya gejala yang dimiliki oleh penyakit tersebut di *knowledge-based*.

Setelah mendapatkan nilai *probability* untuk setiap penyakit, kemudian dilakukan pengurutan secara menurun (*descending*) untuk kemungkinan penyakit yang diderita oleh *user*. Sehingga secara keseluruhan proses dari *forward chaining* mencakup proses *input* gejala *user*, perhitungan nilai kesesuaian sampai perhitungan nilai *probability* untuk setiap penyakit. Untuk lebih jelasnya dapat dilihat pada *flowchart* pada Gambar 3.1.



Gambar 3.1 Flowchart Mesin Inferensi

C. Proses Perhitungan Nilai Fuzzy

Setelah proses inferensi *forward chaining* dilakukan oleh *user* maka perlu dilakukan perhitungan nilai *fuzzy* untuk memperoleh kemungkinan penyakit yang diderita oleh *user*. Pada perhitungan nilai *fuzzy* terdiri atas dua bagian yaitu, pertama perhitungan nilai kesesuaian tiap-tiap gejala untuk suatu penyakit dan yang kedua adalah perhitungan nilai *fuzzy conditional probability* suatu penyakit berdasarkan hasil *inputan* gejala dari *user*.

D. Perhitungan Nilai Kesesuaian

Misalnya jika U adalah suatu *knowledge-based* gejala suatu *set* dari penyakit yang dinyatakan sebagai sebuah *fuzzy set* terhadap gejala A dan B adalah gejala yang diinputkan oleh *user* yang dinyatakan sebagai suatu *fuzzy set* terhadap A, dimana $A = \{a_1, a_2, a_3, \dots, a_n\}$ sedangkan $U = \{\mu_{Uj}(a_1)/(a_1), \mu_{Uj}(a_2)/(a_2), \mu_{Uj}(a_3)/(a_3), \mu_{Uj}(a_4)/(a_4)\}$ dan $B = \{\mu_B(a_1)/(a_1), \mu_B(a_2)/(a_2), \mu_B(a_3)/(a_3), \mu_B(a_4)/(a_4)\}$. Untuk mencari nilai kesesuaian antara *fuzzy set* U dengan B maka dicari seberapa besar selisih antara $\mu_{Uj}(a_1)$ yang merupakan nilai *fuzzy set* gejala a_1 menurut *knowledge-based* dengan $\mu_B(a_1)$ yang merupakan nilai *fuzzy set* gejala a_1 yang diinputkan oleh *user* dibagi dengan nilai $\mu_{Uj}(a_1)$. Jadi rumus untuk mencari nilai kesesuaian *fuzzy set* adalah sebagai berikut:

$$R(B(ai), Uj(ai)) = \text{Max} \left(0,1 - \frac{C|\mu_B(ai) - \mu_{Uj}(ai)|}{\mu_{Uj}(ai)} \right)$$

dengan syarat $\mu_{Uj}(ai) \neq 0$ (4.1)

$R(B(ai), Uj(ai)) = 0$ dengan syarat $\mu_{Uj}(ai) = 0$ (4.2)

Dari perhitungan di atas dapat dilihat nilai kesesuaian yang dihasilkan tiap gejala yang diinput *user* terhadap gejala yang ada pada *knowledge-based* untuk setiap penyakit yang memiliki gejala tersebut. Untuk pengembangan aplikasi sistem pakar ini, perhitungan dibagi menjadi dua bagian, yaitu perhitungan nilai kesesuaian untuk frekuensi gejala dan perhitungan nilai kesesuaian untuk intensitas gejala. Sehingga rumus 4.1 dikembangkan lagi menjadi sebagai berikut :

$$R_{\text{freq}}(B(ai), Uj(ai)) = \text{Max} \left(0,1 - \frac{C|\mu_{B_{\text{freq}}}(ai) - \mu_{Uj_{\text{freq}}}(ai)|}{\mu_{Uj_{\text{freq}}}(ai)} \right)$$

Dengan syarat $\mu_{Uj_{\text{freq}}}(ai) \neq 0$ (4.3)

$$R_{\text{int}}(B(ai), Uj(ai)) = \text{Max} \left(0,1 - \frac{C|\mu_{B_{\text{int}}}(ai) - \mu_{Uj_{\text{int}}}(ai)|}{\mu_{Uj_{\text{freq}}}(ai)} \right)$$

Dengan syarat $\mu_{Uj_{\text{int}}}(ai) \neq 0$ (4.4)

E. Perhitungan Nilai Fuzzy Conditional Probability

Setelah perhitungan nilai kesesuaian gejala antara gejala yang berasal dari *user* dengan gejala yang ada pada *knowledge-based*, maka selanjutnya adalah penjumlahan nilai kesesuaian untuk setiap penyakit, selanjutnya mencari nilai *fuzzy conditional probability* untuk setiap penyakit dengan cara membandingkan

jumlah nilai kesesuaian setiap penyakit dengan banyaknya gejala yang dimiliki oleh penyakit tersebut di *knowledge-based*. Jadi rumus untuk mencari nilai *fuzzy conditional probability* adalah sebagai berikut:

$$P(B,U_n) = \frac{R(B(ai),U_j)}{|U_n|} \text{ dengan syarat } |U_n| \neq 0 \dots \dots \dots (4.5)$$

Keterangan rumus 4.5

$$\sum_{i=1}^m \frac{R(B(ai), U_i)}{|U_n|} \text{ dengan syarat } = \text{Hasil penjumlahan nilai kesesuaian gejala pada penyakit yang ke-}j$$

Misalnya dihitung nilai fuzzy conditional probability pada contoh di subbab 4.3.3.1.1:

$$P(B,U_1) = 1/2 = 0,5$$

$$P(B,U_2) = (1+1)/2 = 1$$

$$P(B,U_3) = 0,7/2 = 0,35$$

Dari hasil perhitungan di atas maka dapat disimpulkan bahwa peluang terkena penyakit U_1 adalah 0,5, peluang *user* terkena penyakit U_2 adalah 1, dan peluang *user* terkena penyakit U_3 0,35. Pada penggunaan rumus 4.5 diatas dapat digunakan untuk menghitung satu parameter saja. Dalam aplikasi sistem pakar ini terdapat dua parameter yang digunakan dalam menghitung nilai *fuzzy conditional probability* suatu penyakit, yaitu parameter frekuensi dan intensitas. Sehingga pada rumus berikut digunakan dua parameter:

$$P(B,U_n) = \frac{C_1 * R_{freq}(B(ai),U_j) + C_2 * R_{int}(B(ai),U_j)}{|U_n|a} \text{ dengan syarat } |U_n| \neq 0 \dots \dots \dots (4.6)$$

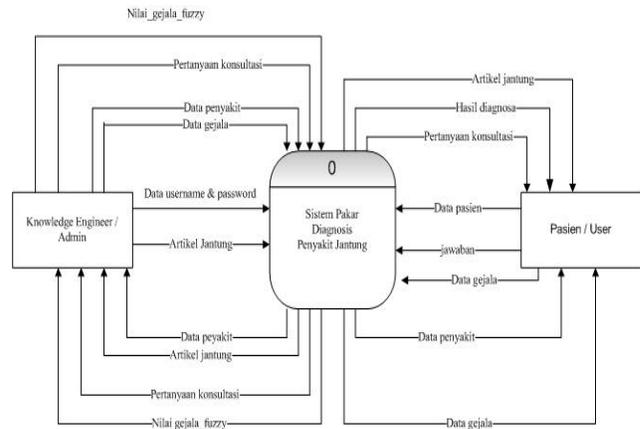
Setelah diperoleh hasil perhitungan *fuzzy conditional probability* masing-masing penyakit, selanjutnya perlu dibuat laporan kemungkinan penyakit yang diderita oleh *user* dengan menggunakan bahasa yang sering digunakan oleh manusia (*variable linguistic*). *Variable linguistic* tersebut antara lain: kecil sekali, kecil, ragu, agak besar, besar, dan hampir pasti. Rentang nilai *fuzzy* untuk masing-masing *variable linguistic* tersebut adalah sebagai berikut:

- Jika nilai_diagnosa ≥ 0.5 dan nilai_diagnosa ≤ 0.599 , maka *variable linguistic*nya "Ragu".
- Jika nilai_diagnosa ≥ 0.6 dan nilai_diagnosa ≤ 0.799 , maka *variable linguistic*nya "Agak besar".
- Jika nilai_diagnosa ≥ 0.8 dan nilai_diagnosa ≤ 0.899 , maka *variable linguistic*nya "Besar".
- Jika nilai_diagnosa ≥ 0.90 , maka *variable linguistic*nya "Hampir pasti".

IV. PERANCANGAN SISTEM

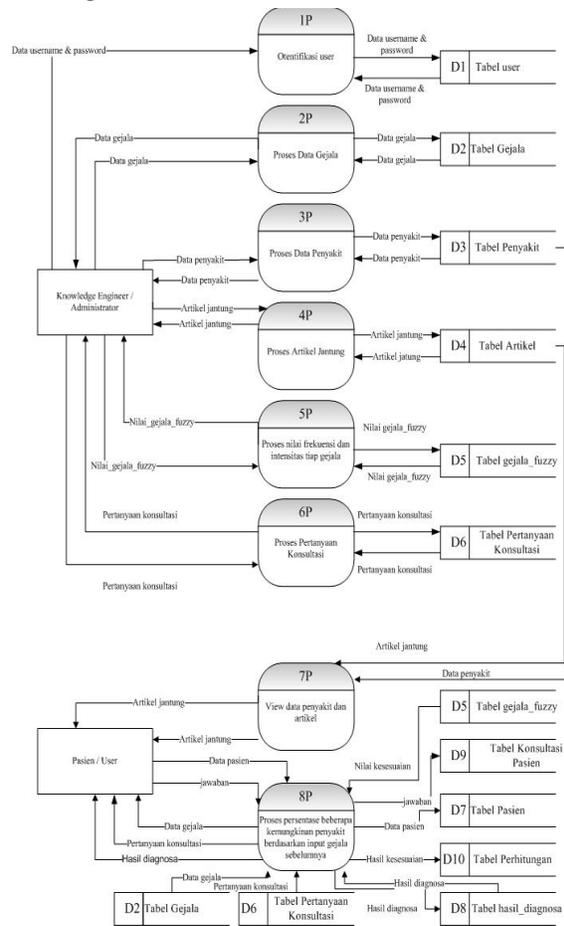
Perancangan DFD dalam aplikasi sistem pakar ini, penulis akan menjelaskan informasi aliran dan transformasi data baik berupa pemasukan data oleh *knowledge engineer* maupun keluaran data yang dapat dilihat oleh *user*. Perancangan sistem ini dimulai dari diagram konteks, dilanjutkan pada DFD level 0, hingga DFD level 1.

A. Diagram konteks



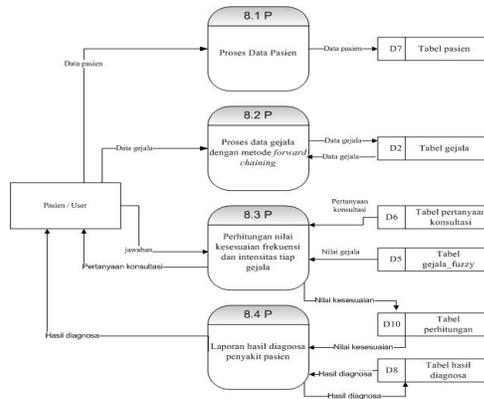
Gambar 4.1 Diagram Konteks

B. Diagram level 0



Gambar 4.2 Diagram level 0

C. Diagram level 1



Gambar 4.3 Diagram level 1

D. Perancangan Basis Data

Beberapa tabel data yang akan diusulkan dalam perancangan sistem pakar ini adalah sebagai berikut:

TABLE I. TABEL ARTIKEL JANTUNG

No.	Field	Type
1	<u>Id</u>	int (20)
2	Judul	varchar (200)
3	headline	Text
4	Isi	Text

TABLE II. TABEL PENYAKIT

No.	Field	Type
1	<u>kode_penyakit</u>	varchar (6)
2	nama_penyakit	varchar (50)
3	Keterangan	Text

TABLE III. TABEL GEJALA

No.	Field	Type
1	<u>kode_gejala</u>	varchar (6)
2	nama_gejala	varchar (50)

TABLE IV. TABEL GEJALA FUZZY

No.	Field	Type
1	<u>kode_penyakit</u>	varchar (6)
2	<u>kode_gejala</u>	varchar (6)
3	intensitas_fuzzy	Double
4	frekuensi_fuzzy	Double

TABLE V. TABEL HASIL DIAGNOSA

No.	Field	Type
1	<u>kode_pasien</u>	varchar (8)
2	<u>kode_penyakit</u>	varchar (6)
3	nilai_akhir	Double
4	variabel_linguistik	varchar (30)

TABLE VI. TABEL KONSULTASI PASIEN

No.	Field	Type
1	<u>kode_pasien</u>	varchar (6)
2	<u>kode_gejala</u>	varchar (6)
3	<u>jawaban_frekuensi</u>	Double
4	<u>jawaban_intensitas</u>	Double

TABLE VII. TABEL 4.17 TABEL DATA PASIEN

No.	Field	Type
1	<u>kode_pasien</u>	varchar (6)
2	Nama	varchar (200)
3	jenis_kelamin	varchar (10)
4	Umur	int (3)
5	Alamat	varchar (50)

TABLE VIII. TABEL PERTANYAAN KONSULTASI

No.	Field	Type
1	<u>kode_gejala</u>	varchar (6)
2	<u>pertanyaan_frekuensi</u>	varchar (200)
3	<u>pertanyaan_intensitas</u>	varchar (200)
4	<u>j_frekuensi_1</u>	varchar (30)
5	<u>j_frekuensi_2</u>	varchar (30)
6	<u>j_frekuensi_3</u>	varchar (30)
7	<u>j_frekuensi_4</u>	varchar (30)
8	<u>j_intensitas_1</u>	varchar (30)
9	<u>j_intensitas_2</u>	varchar (30)
10	<u>j_intensitas_3</u>	varchar (30)
11	<u>j_intensitas_4</u>	varchar (30)

TABLE IX. TABEL PERHITUNGAN

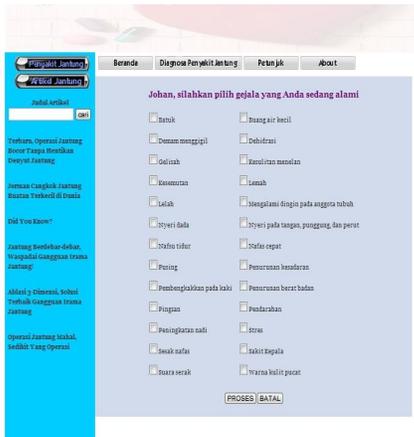
No.	Field	Type
1	<u>kode_pasien</u>	varchar (6)
2	<u>kode_gejala</u>	varchar (6)
3	<u>kode_penyakit</u>	varchar (6)
4	kesesuaian_frekuensi	Double
5	kesesuaian_intensitas	Double
6	jumlah	Double

TABLE X. TABEL DATA USER

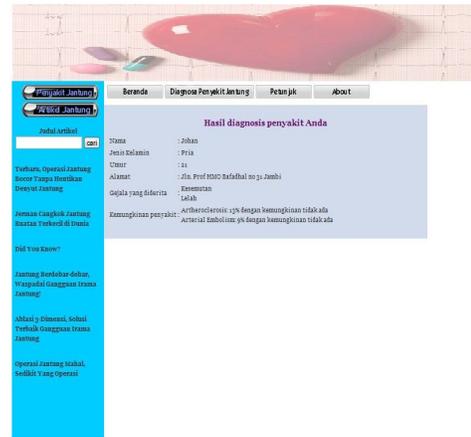
No.	Field	Type
1	<u>Id</u>	int (20)
2	Username	varchar (14)
3	Password	varchar (14)

E. Perancangan Antarmuka

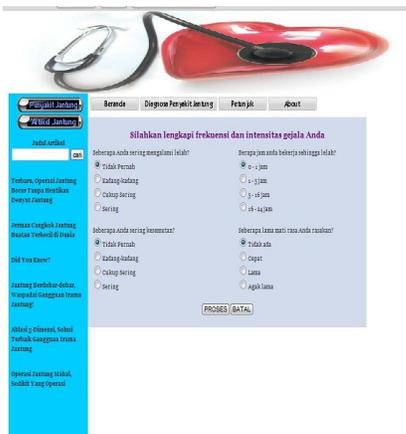
Pada perancangan antarmuka ini penulis menjabarkan dalam desain perancangan *output* dan *input*.



Gambar 4.4 Rancangan *Input* Pemilihan Gejala-gejala penyakit



Gambar 4.8 Rancangan *Output* Hasil Diagnosa Penyakit Pasien



Gambar 4.4 Rancangan *Input* nilai Frekuensi dan Intensitas Gejala

V. KESIMPULAN

Dari analisa dan pembahasan yang telah dilakukan bahwa dengan adanya sistem ini *user* dapat dengan mudah mengetahui penyakit jantung yang diderita penderita dan dengan mudah mendapatkan penjelasan tentang penyakit apa yang dideritanya. Sehingga penggunaan parameter frekuensi dan intensitas gejala sangat bermanfaat untuk proses diagnosa penyakit yang lebih spesifik. Dengan penggunaan metode *fuzzy set* dalam sistem pakar berbasis *web* ini sangat menentukan hasil akhir diagnosa penyakit jantung yang memiliki tingkat keakuratan yang baik. Sehingga sistem pakar ini diharapkan mengurangi angka kematian.

REFERENCES

- [1] Josua Jaya, Frendy dan Saliwijaya Yuki, 2011. *Perancangan Sistem Pakar untuk Mendeteksi Penyakit DBD dan Cara Pengobatan dengan Menggunakan Gadget Berbasis Android*. Universitas Binus. Hal. 23 – 27.
- [2] Arhami, Muhammad, 2005, *Konsep Dasar Sistem Pakar*, Yogyakarta: CV. ANDI OFFSET.
- [3] Tim Penerbit Andi, 2003, *Pengembangan Sistem Pakar Menggunakan Visual Basic*, Yogyakarta: CV. ANDI OFFSET.
- [4] Aditya, Alan Nur, 2011, *Jago PHP & MySQL*, Bekasi: Dunia Komputer.
- [5] Zakaria, Teddy Marcus dan Prijono, Agus, 2007, *Perancangan Antarmuka untuk Interaksi Manusia dan Komputer*, Bandung: Informatika Bandung.
- [6] Lisa, 2010. *Perancangan Sistem Pakar untuk Diagnosis Kerusakan Hardware Laptop Berbasis Web*. Universitas Stikom Dinamika Bangsa Jambi. Hal. 32 - 40.
- [7] Kasron, 2012. *Buku Ajar Gangguan Sistem Kardiovaskuler*. Yogyakarta: Yulia Medika.
- [8] Yahya dan A.Fauzi, FIHA, 2011. *Menaklukkan Pembunuh No.1: Mencegah dan Mengatasi Penyakit Jantung Koroner secara Tepat dan Cepat*. Bandung: PT. Mizan Pustaka.
- [9] Indrasari, ristiana, 2011 *Sistem Pakar Untuk Mendiagnosa Penyakit Autis Berbasis Web*, Semarang: fakultas sistem informasi.