



## Sistem Pakar Mendiagnosa Penyakit Pinched Nerve Menerapkan Metode Rule Base Reasoning

Ratija

Program Studi Teknik Informatika, STMIK Budi Darma, Medan, Indonesia

Email: ratijasaja@gmail.com

### INFORMASI ARTIKEL

Sejarah Artikel:  
 Diterima Redaksi : 07 Juli 2021  
 Revisi Akhir : 15 Juli 2021  
 Diterima : 20 Juli 2021  
 Diterbitkan Online : 28 Juli 2021

### KATA KUNCI

Kata kunci : Saraf Terjepit, Sistem Pakar, Rule Base Reasoning

### KORESPONDENSI

E-mail: ratijasaja@gmail.com

### ABSTRACT

Dengan kemajuan teknologi saat ini, pengetahuan dokter dapat diimplementasikan menjadi sebuah sistem yang disebut sistem pakar. Aplikasi sistem pakar ini dibuat untuk mendiagnosa penyakit pinched nerve/saraf terjepit dengan berbasis android. Susunan sistem saraf manusia tersusun dari sistem saraf pusat dan sistem saraf tepi, saraf memiliki peranan penting dalam tubuh. melalui saraf, komunikasi antara tubuh dan otak berjalan lancar. Sistem pakar adalah sistem berbasis komputer yang menggunakan pengetahuan, fakta, dan teknik penalaran dalam memecahkan masalah yang biasanya hanya dapat dipecahkan oleh seorang pakar dalam bidang tertentu. Sistem pakar memberikan nilai tambah pada teknologi untuk membantu dalam menangani era informasi yang semakin canggih.

Tahap pengembangan aplikasi meliputi analisa perancangan, analisa sistem, implementasi, dan pengujian. Aplikasi Sistem Pakar ini menghasilkan keluaran berupa kemungkinan Saraf terjepit pada tubuh. Sistem ini juga menampilkan besarnya kemungkinan adanya Saraf terjepit pada tubuh. Besarnya nilai tersebut merupakan hasil perhitungan dengan menggunakan metode Rule Base Reasoning.

## 1. PENDAHULUAN

Sistem Pakar (expert system) adalah sistem yang berusaha mengadopsi pengetahuan manusia ke komputer, agar komputer dapat menyelesaikan masalah seperti yang biasa dilakukan oleh para ahli sistem pakar yang baik dirancang agar dapat menyelesaikan suatu permasalahan tertentu dengan meniru kerja para ahli. Sistem pakar atau Expert System biasa disebut juga dengan “knowledge based system” yaitu suatu aplikasi komputer yang ditujukan untuk membantu pengambilan keputusan atau pemecahan persoalan dalam bidang yang spesifik.[1] Pinched nerve atau dalam bahasa indonesia dikenal dengan sebutan saraf terjepit adalah suatu kondisi di mana saraf tertekan oleh bagian sekitarnya. Ketika anda mengalami saraf terjepit, tubuh anda akan mengirimkan sinyal berupa rasa nyeri. Secara konvensional pencegahan penyakit ini dapat dilakukan dengan cara mengompres pada bagian yang sakit, memijatnya dengan lembut dan kemoterapi dengan menggunakan pengawasan dokter, menghindari gerakan – gerakan yang berlebihan.[2]

Adapun masalah yang yang ditimbulkan yaitu banyaknya masyarakat yang terkena penyakit saraf terjepit/pinched nerve. Masyarakat juga sulit menjumpai dokter spesialis saraf dikarenakan keterbatasan waktu dokter spesialis membuat bisa konsultasi dengan dokter spesialis saraf, maka dari itu diperlukan suatu. Metode Rule Based Reasoning salah satu metode dalam sistem pakar yang menggunakan aturan – aturan untuk menyajikan pengetahuannya. Namun jika aturan terlalu banyak, pemeliharaan sistem akan rumit dan terdapat banyak kesalahan dalam kerjanya. Metode ini menggunakan tehnik yang sederhana, dimulai dengan dasar aturan yang berisi semua pengetahuan dari permasalahan yang dihadapi yang kemudian dikodekan ke dalam aturan if-then yang mengandung data, pernyataan dan informasi awal.[3]

Masyarakat indonesia khususnya cenderung untuk memeriksa gigi dan mulut ketika sudah mempunyai sakit yang parah dan mengganggu aktivitas. Untuk itu dibuatlah sistem pakar yang dapat mendeteksi penyakit gigi dan mulut. Pengertian sistem pakar adalah sebuah sistem yang bekerja layaknya seorang ahli dibidangnya sehingga dapat membantu permasalahan dalam hidup. Sistem ini menggunakan metode hybrid rule base-case based reasoning.[3]

## 2. METODOLOGI PENELITIAN

### 2.1 Feature Driven Development (FDD)

Sistem Pakar (expert system) adalah sistem yang berusaha mengadopsi pengetahuan manusia ke komputer, agar komputer dapat menyelesaikan masalah seperti yang biasa dilakukan oleh para ahli. Sistem Pakar yang baik dirancang agar dapat menyelesaikan suatu permasalahan tertentu dengan meniru kerja para ahli. Dengan Sistem

Pakar ini, orang awam juga dapat menyelesaikan masalah yang cukup rumit yang sebenarnya hanya dapat diselesaikan dengan bantuan para ahli. Bagi para ahli, sistem pakar ini juga akan membantu aktivitasnya sebagai asisten yang sangat berpengalaman. Dengan bantuan sistem pakar seseorang yang bukan pakar atau ahli dapat menjawab pertanyaan, menyelesaikan masalah serta mengambil keputusan yang biasanya dilakukan oleh seorang pakar. [1] Sistem pakar akan memberikan pemecahan suatu masalah yang didapat dari dialog dengan pengguna, menyelesaikan masalah serta mengambil keputusan yang biasanya dilakukan oleh seorang pakar. Sistem pakar (expert system) mulai dikembangkan pada pertengahan tahun 1960-an oleh Artificial Intelligence Corporation. Sistem pakar yang muncul pertama kali adalah General-purpose Problem Solver (GPS) yang merupakan sebuah predecessor untuk menyusun langkah-langkah yang dibutuhkan untuk mengubah situasi awal menjadi state tujuan yang telah ditentukan sebelumnya dengan menggunakan domain masalah yang kompleks. [1]

Sistem pakar merupakan program yang dapat menggantikan keberadaan seorang pakar. Alasan mendasar mengapa sistem pakar dikembangkan menggantikan seorang pakar adalah sebagai berikut :

1. Dapat menyediakan kepakaran setiap waktu dan di berbagai lokasi.
2. Secara otomatis mengerjakan tugas-tugas rutin yang membutuhkan seorang pakar.
3. Seorang pakar akan pensiun atau pergi.
4. Menghadirkan atau menggunkan jasa seorang pakar memerlukan biaya yang mahal.
5. Kepakaran dibutuhkan juga pada lingkungan yang tidak bersahabat (hostile environment).

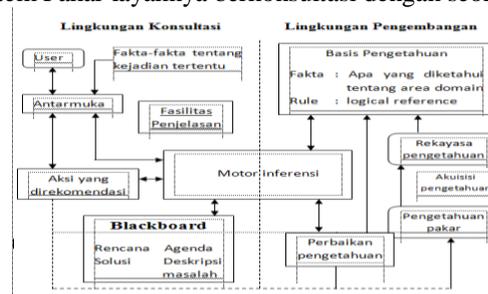
**2.1.1 Konsep Dasar Sistem Pakar**

Pengetahuan dari suatu sistem pakar mungkin dapat direpresentasikan dalam sejumlah cara. Salah satu metode yang paling umum untuk merepresentasikan pengetahuan adalah dalam bentuk tipe aturan (rule) IF..Then (Jika..maka). Walaupun cara diatas sangat sederhana, namun banyak hal yang berarti dalam membangun sistem pakar dengan mengekspresikan pengetahuan pakar dalam bentuk aturan diatas. Konsep dasar dari suatu sistem pakar mengandung beberapa unsur atau elemen yaitu :[2].

1. Keahlian  
Keahlian merupakan suatu penguasaan pengetahuan dibidang tertentu yang didapatkan dari pelatihan, membaca atau pengalaman.
2. Ahli  
Seorang ahli adalah seorang yang mampu menjelaskan suatu tanggapan, mempelajari hal-hal baru seputar topik permasalahan (domain), menyusun kembali pengetahuan, memecah aturan-aturan jika diperlukan dan menentukan relevan tidaknya keahlian mereka.
3. Pengalihan keahlian  
Pengalihan keahlian dari para ahli ke komputer untuk kemudian dialihkan lagi ke orang lain yang bukan ahli (tujuan utama sistem pakar). Proses ini membutuhkan 4 aktivitas, yaitu: tambahan pengetahuan (dari para ahli atau sumber-sumber lainnya), representasi pengetahuan yang berupa fakta dan prosedur (ke komputer), inferensi pengetahuan dan pengalihan pengetahuan ke pengguna.
4. Inferensi  
Mekanisme inferensi merupakan perangkat lunak yang melakukan penalaran dengan menggunakan pengetahuan yang ada untuk menghasilkan suatu kesimpulan atau hasil akhir.
5. Aturan  
Aturan merupakan informasi tentang cara bagaimana memperoleh fakta baru dari fakta yang telah diketahui.
6. Kemampuan menjelaskan.  
Kemampuan komputer untuk memberikan penjelasan kepada pengguna tentang sesuatu informasi tertentu dari pengguna dan dasar yang dapat digunakan oleh komputer untuk dapat menyimpulkan suatu kondisi.

**2.1.2 Struktur Sistem Pakar**

Ada dua bagian penting dari sistem pakar, yaitu lingkungan pengembangan (development environment) dan lingkungan konsultasi (consultation environment). Lingkungan pengembangan digunakan oleh pembuat sistem pakar untuk membangun komponen-komponennya dan memperkenalkan pengetahuan kedalam knowledge base (basis pengetahuan). Lingkungan konsultasi digunakan oleh pengguna untuk berkonsultasi sehingga pengguna mendapatkan pengetahuan dan nasihat dari Sistem Pakar layaknya berkonsultasi dengan seorang pakar. [2]



**Gambar 2.1** Komponen – Komponen yang Penting dalam Sebuah Sistem Pakar  
Sumber : Muhammad Arhami, konsep dasar sistem pakar, 2005 [2]

### 3. ANALISA DAN PEMBAHASAN

#### 3.1 Analisa

Analisa berguna untuk mengetahui kebutuhan perangkat lunak serta kebutuhan sistem pakar yang dibangun. Dalam tahap ini dilakukan pencarian dan pengumpulan data serta pengetahuan yang diperlukan oleh sistem pakar. Sehingga pada akhirnya didapatkan hasil dari analisa berupa sebuah sistem yang strukturnya dapat didefinisikan dengan baik dan jelas. Tujuan dari analisa sistem dalam pengembangan sistem pakar ini adalah untuk mendapatkan berbagai kebutuhan pengguna dan sistem, yaitu perihal masukan dan keluaran yang harus disediakan oleh pengguna serta dibutuhkan oleh sistem. Proses tersebut akan menjadi masukan bagi proses perancangan sistem secara keseluruhan. Program sistem pakar merupakan program dengan basis pengetahuan yang dinamis. Dengan kata lain, pengetahuan yang ada pada program ini harus dapat bertambah atau bisa di edit tanpa harus mengubah isi dari program secara keseluruhan. Dengan demikian diperlukan pengetahuan yang akuisisi menjadi basis pengetahuan dengan mengetahui terlebih dahulu bagaimana mengenali suatu jenis penyakit yang diderita seseorang berdasarkan keluhan utama maupun gejala – gejala yang ditimbulkan serta sebab terjadinya penyakit. Untuk memenuhi syarat – syarat tersebut maka dibuat suatu struktur *if-then* yang dibaca dari database. Program ini diusahakan sesederhana mungkin agar mudah dimengerti pengguna.

Rule Base Reasoning (*RBR*) merupakan aturan logis dimana setiap aturannya di dapat dari studi literatur dan informasi dari ahli tanpa melihat kasus yang dihadapi. beberapa cara alternatif untuk memperoleh aturan menggunakan metode pembelajaran mesin berdasarkan data empiris yang ada. Metode ini menggunakan tehnik sederhana, dimulai dengan dasar aturan yang berisi semua pengetahuan dari permasalahan yang dihadapi yang kemudian di kodekan kedalam aturan *if-then* yang mengandung data pernyataan. Satu aturan di representasikan dengan : IF <kondisi> THEN <kesimpulan>, dimana setiap kondisi-kondisi dari aturan ke aturan yang lainnya terhubung satu dengan yang lain melalui penghubung logika seperti penghubung dan, atau, negasi, serta penghubung lainnya membentuk sebuah fungsi logis.

Dari sumber yang dapat dipercaya maka diperoleh beberapa gejala – gejala saraf terjepit adalah:

**Tabel 3:** Gejala Saraf terjepit yang dialami pasien

Kode_Gejala	Gejala
G01	Nyeri Kaki
G02	Pembengkakan
G03	Mudah Kesemutan
G04	Nyeri Paha
G05	Otot Lemah
G06	Kelumpuhan
G07	Nyeri Leher
G08	Leher Kaku
G09	Nyeri Pinggang
G010	Susah Buang air Kecil
G011	Mati Rasa
G012	Nyeri punggung

Sumber : Konsultasi Dr. Kolman Saragih S.Ps

**Tabel 3.2:** Kondisi saraf terjepit

No.	Kondisi Saraf terjepit
1	Saraf terjepit pada Punggung
2	Saraf Terjepit di Kaki
3	Saraf Terjepit di Paha
4	Saraf Terjepit pada leher
5	Saraf terjepit di pinggang

#### 3.1 Analisa Rule Based Reasoning

Adapun analisa terhadap sistem pakar yang dibangun merupakan rule yang menerapkan metode Rule Base Reasoning. Teori Rule Base Reasoning merupakan suatu metode ketidakpastian berupa teori matematis untuk membuktikan fungsi kepercayaan dan penalaran yang masuk akal yang digunakan untuk mengklasifikasikan beberapa informasi berupa bukti kemungkinan dari suatu kejadian. Adapun rumus untuk menghitung kemiripan antara kasus baru dan kasus lama adalah sebagai berikut :

$$\text{Similarity} = \frac{S1 * W1 + S2 * W2 + \dots + Sn * Wn}{W1 + W2 + \dots + Wn}$$

keterangan :

Similarity = (nilai kemiripan)

W = weight (bobot yang diberikan)

### 3.1.1 Penerapan Metode Rule Base Reasoning

Ada beberapa gejala dari penyakit syaraf terjepit. Berikut ini adalah contoh perhitungan secara manual dengan menggunakan metode *Rule Based Reasoning* (RBR) dan ada beberapa cara penyelesaian metode *Rule Based Reasoning* adalah sebagai berikut :

#### 1. Tahap *retrieve*

Pencarian kemiripan antara kasus baru dan lama dilakukan dengan cara mencocokkan gejala yang di inputkan pada halaman konsultasi oleh pengguna dengan gejala yang ada pada basis pengetahuan setelah itu baru menghitung kemiripan antara kasus menggunakan rumus *rule base reasoning*.

**Tabel 3.3** Gejala Saraf terjepit yang dialami pasien

Kode_Gejala	Gejala
G01	Nyeri Kaki
G02	Pembengkakan
G03	Mudah Kesemutan
G04	Nyeri Paha
G05	Otot Lemah
G06	Kelumpuhan
G07	Nyeri Leher
G08	Leher Kaku
G09	Nyeri Pinggang
G010	Susah Buang air Kecil
G011	Mati Rasa
G012	Nyeri punggung

**Tabel 3.4:** Kondisi saraf terjepit

No.	Kondidi Saraf terjepit
1	Saraf terjepit pada Punggung
2	Saraf Terjepit di Kaki
3	Saraf Terjepit di Paha
4	Saraf Terjepit pada leher
5	Saraf terjepit di pinggang

Kemudian sistem mencari kemiripan antara gejala yang di inputkan pengguna dengan gejala yang berada pada basis pengetahuan. Setelah itu menghitung nilai kemiripannya. Berikut ini penyakit yang memiliki yang sama dengan gejala pengguna :

**Tabel 3.5:** Gejala Saraf terjepit yang dialami pasien

Kode_Gejala	Gejala
G01	Nyeri Kaki
G02	Pembengkakan
G03	Mudah Kesemutan
G04	Nyeri Paha
G05	Otot Lemah

G06	Kelumpuhan
G07	Nyeri Leher
G08	Leher Kaku
G09	Nyeri Pinggang
G010	Susah Buang air Kecil
G011	Mati Rasa
G012	Nyeri punggung

**Tabel 3.6:** Gejala Saraf Terjepit Di Punggung

Kode-gejala	Gejala
G1	Nyeri Punggung
G2	Punggung kaku
G3	Bantalan tulang belakang menonjol
G4	Mudah kesemutan

**Tabel 3.7** Saraf Terjepit Di Punggung

Kode_gejala	Nama gejala	Boot	Nilai kemiripan	Bobot x nilai
G3	Nyeri punggung	3	1	3
G6	Punggung kaku	3	0	0
G7	Bantalan tulang belakang menonjol	3	0	0
G5	Mudah Kesemutan	5	1	5
JUMLAH		14		8

$$\text{Similarity} = \frac{(1 \cdot 3) + (0 \cdot 3) + (0 \cdot 3) + (1 \cdot 5)}{3+3+3+5} = \frac{3+5}{14} = \frac{8}{14} = 0,57$$

**Tabel 3.8:** Gejala Saraf terjepit yang dialami pasien

Kode_Gejala	Gejala
G01	Nyeri Kaki
G02	Pembengkakan
G03	Mudah Kesemutan
G04	Nyeri Paha
G05	Otot Lemah
G06	Kelumpuhan
G07	Nyeri Leher
G08	Leher Kaku
G09	Nyeri Pinggang

G010	Susah Buang air Kecil
G011	Mati Rasa
G012	Nyeri punggung

**Tabel 3.9** Gejala Saraf Terjepit Di Kaki

Kode_gejala	Gejala
G1	Nyeri kaki
G5	Mudah kesemutan
G16	Pembengkakan
017	Otot melemah

**Tabel 3.10** Syaraf Terjepit Di Kaki

Kode_gejala	Nama gejala	Bobot	Nilai kemiripan	Bobot x nilai
G3	Nyeri kaki	3	1	3
G6	Mudah kesemutan	3	1	3
G7	Pembengkakan	3	1	3
G8	Otot menonjol	5	0	0
JUMLAH		9		9

$$\text{Similarity} = \frac{(1 \cdot 3) + (1 \cdot 3) + (1 \cdot 3) + (0 \cdot 3)}{3+3+3+5} = \frac{3+3+3+0}{14} = \frac{9}{14} = 0,64$$

**Tabel 3.11** Gejala Saraf terjepit yang dialami pasien

Kode_Gejala	Gejala
G01	Nyeri Kaki
G02	Pembengkakan
G03	Mudah Kesemutan
G04	Nyeri Paha
G05	Otot Lemah
G06	Kelumpuhan
G07	Nyeri Leher
G08	Leher Kaku
G09	Nyeri Pinggang
G010	Susah Buang air Kecil
G011	Mati Rasa
G012	Nyeri punggung

**Tabel 3.12** Gejala saraf terjepit di Paha

Kode_gejala	Gejala
G5	Nyeri Paha
G9	Mudah Kesemutan
G10	Otot Paha Lemah
G11	Kelumpuhan
G12	Mati rasa

**Tabel 3.13** Saraf terjepit di Paha

Kode_gejala	Nama gejala	Bobot	Nilai kemiripan	Bobot x nilai
G5	Mudah kesemutan	3	1	3
G9	Nyeri paha	3	1	3
G10	Matai rasa	3	1	3
G11	Kelumpuhan	3	1	3
G12	Otot paha lemah	5	0	0
JUMLAH		17		12

$$\text{Similarity} = \frac{(1 * 3) + (1 * 3) + (1 * 3) + (1 * 3) + (5 * 0) - 3 + 3 + 3 + 3 - 12}{3 + 3 + 3 + 3 + 5} = \frac{17 - 12}{17} = 0,70$$

**Tabel 3.14** Gejala Saraf terjepit yang dialami pasien

Kode_Gejala	Gejala
G01	Nyeri Kaki
G02	Pembengkakan
G03	Mudah Kesemutan
G04	Nyeri Paha
G05	Otot Lemah
G06	Kelumpuhan
Kode_Gejala	Gejala
G07	Nyeri Leher
G08	Leher Kaku
G09	Nyeri Pinggang
G010	Susah Buang air Kecil
G011	Mati Rasa
G012	Nyeri punggung

**Tabel 3.15** Gejala Saraf Terjepit Pada Leher

Kode_gejala	Gejala
G7	Mati rasa
G9	Leher kaku

G10 Mati rasa

**Tabel 3.16** Saraf terjepit pada leher

Kode_gejala	Nama gejala	Bobot	Nilai kemiripan	Bobot x nilai
G7	Mati rasa	3	1	3
G9	Nyeri leher	3	1	3
G10	Leher otot melemah	5	0	0
JUMLAH		11		6

$$\text{Similarity} = \frac{(1 \times 3) + (1 \times 3) + (0 \times 5)}{3+3+5} = \frac{3+3}{11} = \frac{6}{11} = 0,54$$

**Tabel 3.17** Gejala Saraf terjepit yang dialami pasien

Kode_Gejala	Gejala
G01	Nyeri Kaki
G02	Pembengkakan
G03	Mudah Kesemutan
G04	Nyeri Paha
G05	Otot Lemah
G06	Kelumpuhan
G07	Nyeri Leher
G08	Leher Kaku
G09	Nyeri Pinggang
G010	Susah Buang air Kecil
G011	Mati Rasa
G012	Nyeri punggung

**Tabel 3.18** Gejala saraf serjepit di Pinggang

Kode_gejala	Gejala
G7	Mati rasa
G6	Melemah
G9	Nyeri pinggang
G21	Susah buang air kecil
G5	Mudah kesemutan

**Tabel 3.19** Saraf Terjepit Di Pinggang

Kode_gejala	Nama gejala	Bo bot	Nilai kemiripan	Bobot x nilai
G7	Mati rasa	3	1	3
G6	Melenah	3	0	0

G9	Nyeri pinggang	3	1	3
G5	Mudah kesemutan	3	1	3
G21	Susah buang air kecil	3	1	3
<b>JUMLAH</b>		<b>15</b>		<b>12</b>

$$\text{Similarity} = \frac{(1 * 3) + (0 * 3) + (1 * 3) + (1 * 3) + (1 * 3)}{3+3+3+3+3} = \frac{3+3+3+3+3}{15} = \frac{12}{15} = 0,8$$

2. Tahap *Reuse*

Berdasarkan contoh kasus di atas penyakit yang memiliki gejala sama dengan gejala yang di inputkan oleh pengguna adalah:

1. Saraf terjepit di punggung, nilai *similarity* = 0,57
2. Saraf terjepit di kaki, nilai *similarity* = 0,64
3. Saraf terjepit di paha, nilai *similarity* = 0,70
4. Saraf terjepit di leher, nilai *similarity* = 0,54
5. Saraf terjepit di pinggang, nilai *similarity* = 0,8

Pada tahap reuse solusi yang diberikan adalah solusi yang memiliki nilai kemiripan tertinggi antara kasus lama dengan baru, dalam contoh kasus ini nilai kemiripan yang tinggi terhadap pada penyakit saraf terjepit di paha dengan nilai kemiripan 0,70. Maka pasien terkena penyakit saraf terjepit di paha.

3. Tahap *revise*

Tahap revise adalah tahap peninjauan kembali solusi yang digunakan untuk mengelola data kasus baru untuk dilakukan validasi terhadap kesimpulan yang dimiliki. Kasus yang telah di validasi oleh admin atau pakar kemudian disimpan ke dalam basis pengetahuan untuk digunakan pada kasus selanjutnya

4. Tahap *retain*

Tahap ini menyimpan kasus lama ke dalam basis pengetahuan yang nantinya akan di gunakan untuk memecahkan kasus baru. Pengguna dapat memasukkan gejala dan penyakit yang telah di konsultasikan dengan menekan *retain* setelah itu gejala dan penyakit masuk kedalam tabel retain yang kemudian akan di jadikan kasus selanjutnya.

Maka hasil perhitungan tertinggi dari seluruh yang ada adalah saraf terjepit di paha dengan nilai densitas 0,70. Jadi pasien menderita penyakit saraf terjepit pada bagian paha.

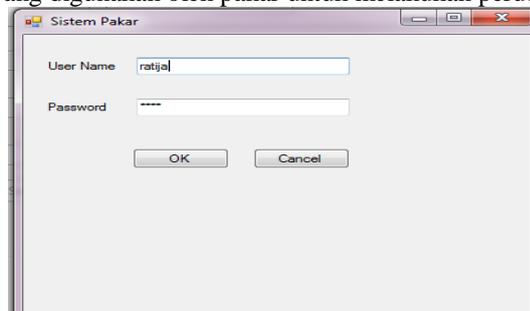
## 4. IMPLEMENTASI

### 4.1 Tampilan Menu Utama

Tahap akhir dari pembuatan perangkat lunak adalah implementasi. Implementasi ini akan ditampilkan dalam bentuk program yang dirancang sebagai berikut:

1. Tampilan *Form Login*

Tampilan *Login* Pada pakar yang digunakan oleh pakar untuk melakukan perubahan pada gejala-gejala.



**Gambar 4.1** Tampilan *Form Login*

2. Tampilan Menu Utama

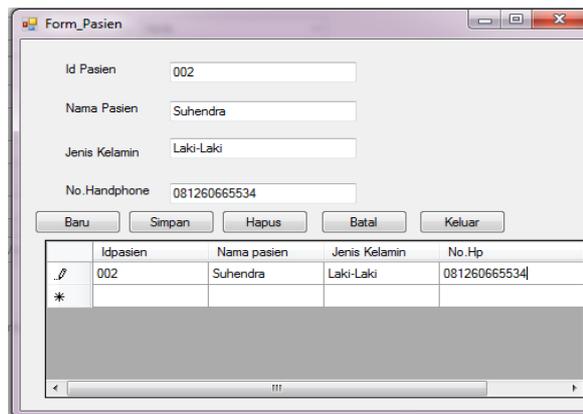
Tampilan Menu utama merupakan tampilan utama setelah *login*, di menu utama terdapat pilihan konsultasi, tentang dan keluar.



**Gambar 4.2** Tampilan Form Menu Utama

3. Tampilan Form Pasien

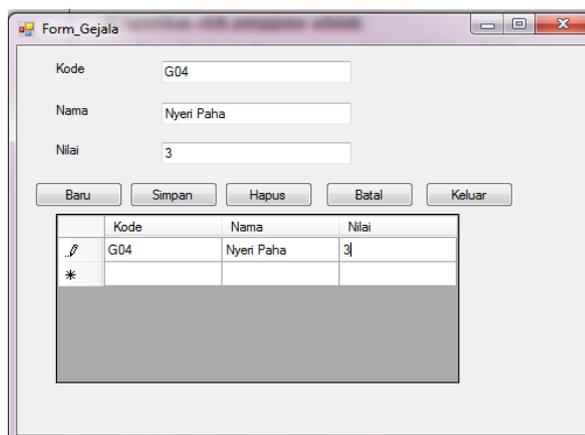
Tampilan Pasien ini merupakan tampilan yang hanya bisa diakses oleh pakar. Ditampilkan inilah dilakukan pengeditan data pasien.



**Gambar 4.4** Tampilan Form pasien

4. Tampilan Form Gejala

Tampilan Gejala ini merupakan tampilan yang hanya bisa diakses oleh pakar. Ditampilkan inilah dilakukan pengeditan gejala.



**Gambar 4.3** Tampilan Form Gejala

5. Tampilan Form Penyakit

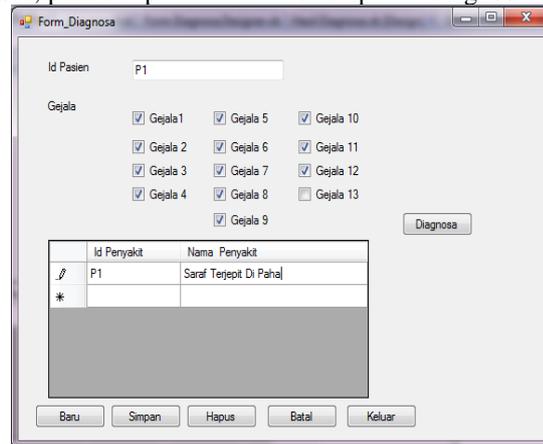
Tampilan Penyakit ini merupakan tampilan yang hanya bisa diakses oleh pakar. Ditampilkan inilah dilakukan pengeditan penyakit.



Gambar 4.5 Tampilan Form Penyakit

6. Tampilan Form Diagnosa

Tampilan ini diakses oleh *user*, pada tampilan ini akan ditampilkan diagnosa.



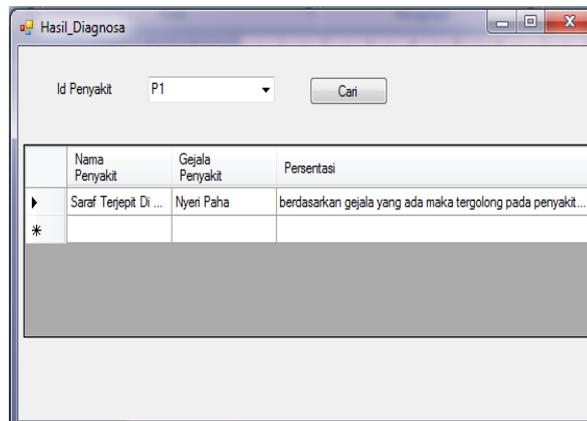
Gambar 4.6 Tampilan Form Diagnosa

4.4 Hasil Pengujian Program

Adapun tampilan hasil pengujian program yang telah dirancang oleh penulis adalah sebagai berikut :

1. Tampilan Form Data Hasil Diagnosa

Tampilan Form Data Hasil Diagnosa adalah tampilan data yang membuat hasil diagnosa gejala dan penyakit saraf terjepit.



Gambar 4.7 Tampilan Form Hasil Diagnosa

**5. KESIMPULAN**

1. Berdasarkan pengembangan yang telah dilakukan selama proses parancangan hingga implementasi system pakar untuk mendiagnosa saraf terjepit dengan menggunakan metode Rule Based Reasoning (RBR), maka dapat diambil kesimpulan sebagai berikut :
2. Dalam mengetahui gejala penyakit saraf terjepit dengan system pakar melihat dari gejala-gejala yang ada dapat dilakukan untuk membantu masyarakat untuk mengetahui saraf terjepit.

3. Dengan menerapkan metode Rule Based Reasoning (RBR) pada penyakit saraf terjepit dapat menghasilkan perhitungan valid yang sama dengan perhitungan manual sehingga proses diagnosa dapat dilakukan dengan cepat dan akurat.
4. Dengan merancang suatu aplikasi sistem pakar berbasis rule based reasoning untuk memudahkan proses konsultasi tentang penyakit saraf terjepit.

## REFERENCES

- [1] sari iswanti sri hartati, SISTEM PAKAR DAN PENBANGUNGANNYA. Yogyakarta 55511: graha ilmu, 2008.
- [2] dr. claire weekes, meredakan ketegangan saraf anda. semarang: dahara prize, 2001.
- [3] siti saadah m. abdurrachman irfandi, ade romadhony, "implementasi sistem pakar diagnosa penyakit gigi dan mulut menggunakan metode hybrid case-based dan rule based reasoning," prodi s1 Tek. Inform. Fak. Tek. Inform. Univ. telkom, 2015.
- [4] naufia oktaviani syamsiah, "penerapan esdlc pada sistem pakar forward chaining dan rule base reasoning untuk dignosa awal penyakit kardiovaskulas dan paru-paru manusia," Tek. Inform. stmik antar bangsa, vol. 1, no. 2224-2444, 2015.
- [5] sari iswanti sri hartati, SISTEM PAKAR DAN PENBANGUNGANNYA. Yogyakarta 55511: graha ilmu, 2008.
- [6] muhammad arhami, konsep dasar sistem pakar. Yogyakarta 55281: ANDI, 2005.
- [7] D. vincent suhartono T.Sutojo, s.si, m.kom., edy mulyanto, s.si., m.kom., kecerdasan buatan. semarang: ANDI, 2011.
- [8] permanan ginting munthe maiyulis, muhammad syahrizal, "sistem pakar mendiagnosa penyakit leptospirosis menerapkan metode rule base reasoning," vol. 13, no. issn 2339-210x, p. 34, 2018.
- [9] Andi Muh. Lukman, "Aplikasi Diagnosa Penyakit Gigi Dan Pengobatannya Menggunakan Metode Forward Chaining Dan Association Rule Analysis," Fak. Tek. UPRI makasar, vol. 3, no. issn 2355-3677, p. 72, 2016.
- [10] dr. claire weekes, meredakan ketegangan saraf anda. semarang: dahara prize, 2001.
- [11] M. shalahuddin Rosa A. S, rekayasa perangkat lunak. bandung: informatika bandung, 2016.
- [12] M. m.Primananda Arif Aditya, s.si, dasar-dasar pemrograman database desktop dengan visual basic.net 2008. jakarta: pt. elex media komputindo, 2013.
- [13] Ir. Hartanto Kristanto, konsep dan perancangan database. Yogyakarta 55281: ANDI, 2004.
- [14] Wahana Komputer, microsoft office 2010. yogyakarta: Andi, 2010.