

BAB II

LANDASAN TEORI

2.1 Kecerdasan Buatan

Sutojo, Mulyanto dan Suhartono (2011:2) menguraikan bahwa kecerdasan buatan berasal dari bahasa inggris “*Artifiacial intellegence*” atau di singkat AI ,yaitu intellegence adalah kata sifat yang berarti cerdas, sedangkan artificial artinya buatan. Kecerdasan yang dimaksud di sini merujuk pada mesin yang mampu berpikir, menimbang tindakan yang akan diambil, dan mampu mengambil keputusan seperti yang di lakukan manusia. Kecerdasan diciptakan dan dimasukan ke dalam suatu mesin (komputer) agar dapat dilakukan manusia, beberapa macam bidang yang menggunakan kecerdasan buatan antara lain sistem pakar, game playing, logika fuzzy, jaringan saraf tiruan (JST) dan robotika.

2.2 Sistem Pakar

2.2.1 Definisi Sistem Pakar

Sutojo, Mulyanto dan Suhartono (2011:159) menguraikan bahwa sistem pakar adalah suatu sistem yang di rancang untuk dapat menirukan keahlian seorang pakar dalam menjawab pertanyaan dan memecahkan suatu masalah. Sistem pakar akan memberikan pemecahan suatu permasalahan yang di dapat dari dialog dengan pengguna. Dengan bantuan sistem pakar seseorang yang bukan pakar atau ahli dapat menjawab pertanyaan, menyelesaikan masalah serta mengambil keputusan yang biasanya dilakukan seorang pakar.

2.2.2 Ciri-Ciri Sistem Pakar

Sutojo (2011:162) menguraikan bahwa ciri-ciri sistem pakar adalah :

1. Terbatas pada domain keahlian tertentu.
2. Dapat memberikan penalaran-penalaran untuk data-data yang tidak lengkap atau tidak pasti.
3. Dapat mengemukakan rangkaian alasan-alasan yang diberikannya dengan cara yang dapat dipahami.

4. Berdasarkan pada kaidah / ketentuan / rule tertentu.
5. Dirancang untuk dapat dikembangkan secara bertahap.
6. Pengetahuan dan mekanisme penalaran (*inference*) jelas terpisah.
7. Keluarannya bersifat anjuran.
8. Sistem dapat mengaktifkan kaidah secara searah yang sesuai dituntun oleh dialog dengan user.

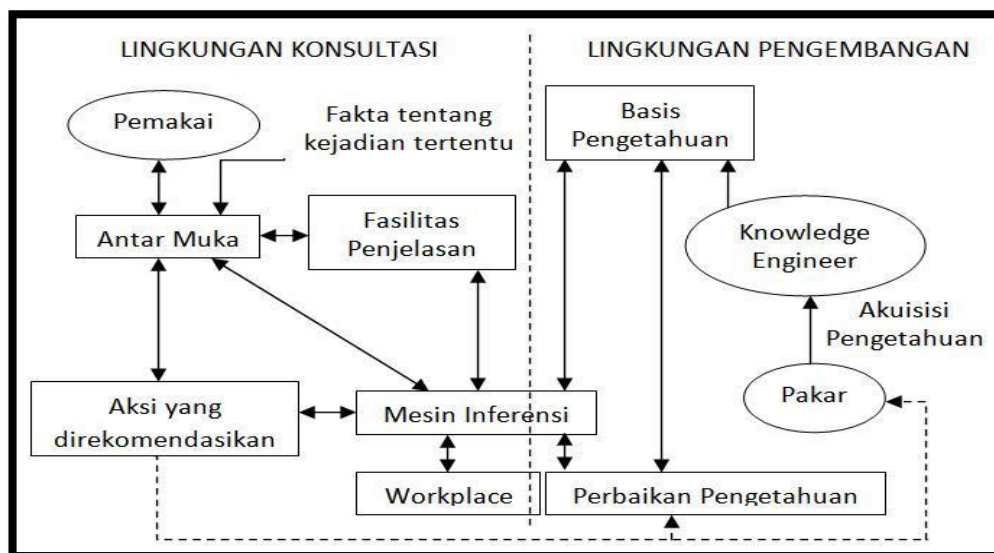
2.2.3 Manfaat Sistem Pakar

Sutojo (2011:160) menguraikan bahwa sistem pakar menjadi populer karena sangat banyak kemampuan dan manfaat yang diberikannya, di antaranya :

1. Meningkatkan produktivitas, karena sistem pakar dapat bekerja lebih cepat dari pada manusia.
2. Membuat seorang yang awam bekerja seperti layaknya seorang pakar.
3. Meningkatkan kualitas, dengan memberi nasihat yang konsisten dan mengurangi kesalahan.
4. Mampu menangkap pengetahuan dan kepakaran seseorang.
5. Dapat beroperasi di lingkungan yang berbahaya.
6. Memudahkan akses pengetahuan seorang pakar.
7. Handal, sistem pakar tidak pernah menjadi bosan dan kelelahan atau sakit.
8. Meningkatkan kapabilitas sistem komputer. Integrasi sistem pakar dengan sistem komputer lain membuat sistem lebih efektif dan mencakup lebih banyak aplikasi.
9. Mampu bekerja dengan informasi yang tidak lengkap atau tidak pasti.
10. Bisa digunakan sebagai media pelengkap dalam pelatihan. Pengguna pemula yang bekerja dengan sistem pakar akan menjadi lebih berpengalaman karena adanya fasilitas penjelas yang berfungsi sebagai guru.

2.2.4 Struktur Sistem Pakar

Sutojo, Mulyanto dan Suhartono (2011:167) menguraikan bahwa sistem pakar disusun oleh dua bagian utama, yaitu bagian lingkungan pengembangan (*development environment*) dan lingkungan konsultasi (*consulation environment*). Lingkungan pengembang digunakan oleh pembuat sistem pakar untuk membangun komponen komponennya dan memperkenalkan pengetahuan kedalam knowledge base (basis pengetahuan). Sedangkan lingkungan konsultasi digunakan oleh pengguna untuk berkonsultasi sehingga pengguna mendapatkan pengetahuan dan nasihat dari sistem pakar layaknya berkonsultasi dengan seorang pakar. Gambar 2.1 di bawah ini menguraikan Struktur dalam Sistem Pakar :



Gambar 2.1 Struktur dalam Sistem Pakar

Struktur yang terdapat dalam Sistem Pakar adalah sebagai berikut:

1). Antarmuka Pengguna

Media komunikasi antara pengguna/ pemakai/ user dengan sistem. Pada bagian antar muka ini akan terjadi dialog antara pengguna dengan sistem pakar dengan tampilan yang dapat dimengerti oleh pengguna.

2.) Basis pengetahuan (*knowledge base*)

Dalam basis pengetahuan terdapat pengetahuan yang diperlukan dalam system.

Ada 2 basis pengetahuan yaitu:

- a. Fakta merupakan kondisi, situasi, atau permasalahan yang ada.
- b. Aturan (*Rule*) merupakan cara menggunakan pengetahuan untuk memecahkan masalah.

3.) Akuisisi pengetahuan

Memasukkan pengetahuan ke dalam sistem. Pengetahuan yang di transfer ke dalam sistem bisa berasal dari seseorang yang ahli dalam bidang tertentu ataupun pengetahuan tersebut bisa berasal dari laporan riset, buku, atau informasi yang valid / bisa dipertanggung jawabkan.

4.) Mesin inferensi (*Interference Engine*)

Berfungsi untuk memperoleh solusi atau menarik kesimpulan berdasarkan basis pengetahuan.

5.) Workplace / Blackboard Area

Untuk merekam hasil sementara yang akan dijadikan sebagai keputusan sementara.

6.) Fasilitas Penjelasan

Berfungsi untuk memberikan penjelasan kepada pengguna tentang cara kerja sistem dalam mengambil suatu kesimpulan. Penjelasan kepada pengguna dapat berbentuk keterangan setelah pertanyaan diajukan.

7.) Perbaikan pengetahuan

Seorang pakar memiliki kemampuan untuk memperbaiki pengetahuan dengan cara belajar dan menganalisis. Perbaikan pengetahuan juga diperlukan dalam sistem pakar agar sistem dapat memberikan alasan sukses dan gagalnya dalam mengambil keputusan dan dapat mengevaluasi pengetahuan yang ada dalam sistem dapat digunakan di masa yang akan datang atau ada perubahan pengetahuan.

8.) Pengguna/ Pemakai

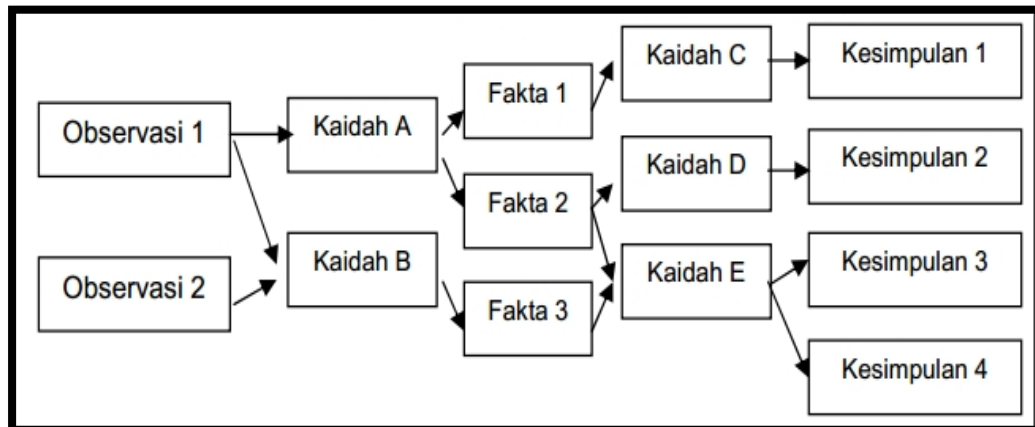
Pengguna adalah seseorang yang menggunakan sistem pakar tersebut untuk mendapatkan pengetahuan, solusi, atau saran dari suatu permasalahan yang berhubungan dengan sistem pakar tersebut.

2.3 Metode Inferensi Dalam Sistem Pakar

Sutojo, Mulyanto dan Suhartono (2011:171) menguraikan bahwa inferensi merupakan proses untuk menghasilkan informasi dari fakta yang diketahui atau diasumsikan. Inferensi adalah konklusi logis (*logical conclusion*) atau implikasi berdasarkan pada informasi yang tersedia. Dalam sistem pakar, proses inferensi dilakukan dalam suatu modul yang disebut Inference Engine (mesin inferensi). Ketika representasi pengetahuan pada bagian knowledge base telah lengkap, atau paling tidak telah berada pada level cukup akurat, maka refresensi pengetahuan tersebut telah siap digunakan. Sedangkan inference engine merupakan modul yang berisi program tentang bagaimana mengendalikan proses reasoning. Sesuai dengan tujuan sistem pakar untuk mengembangkan dan memasyarakatkan serangkaian usulan jawaban dari suatu masalah, untuk itu sistem pakar memiliki suatu strategi penalaran (*inference*) dimana didalam proses penalaran itu akan ditemukan berbagai macam jawaban.

2.3.1 Pelacakan Kedepan (*Forward Chaining*)

Sutojo, Mulyanto dan Suhartono (2011:171) menguraikan bahwa *Forward Chaining* merupakan teknik pencarian yang dimulai dengan fakta yang diketahui, kemudian mencocokkan fakta-fakta tersebut dengan bagian IF dari *rules* IF-THEN. Bila ada fakta yang cocok dengan bagian IF, maka *rule* tersebut dieksekusi. Bila sebuah *rule* dieksekusi, maka sebuah fakta baru (bagian THEN) ditambahkan ke dalam database. Setiap kali pencocokan berhenti bila tidak ada lagi *rule* yang bisa dieksekusi. Metode pencarian yang digunakan adalah *Depth-First Search* (DFS), *Breadth-First Search* (BFS), atau *Best First Search*. Gambar 2.2 berikut ini menguraikan diagram pelacakan kedepan (*Forward Chainig*) :



Gambar 2.2 Proses *Forward Chaining*

Kelebihan dari teknik *Forward Chaining* adalah sebagai berikut :

1. Kelebihan utama dari *Forward Chaining* yaitu metode ini akan bekerja dengan baik ketika problem bermula dari mengumpulkan atau menyatukan informasi lalu kemudian mencari kesimpulan apa yang dapat diambil dari informasi tersebut.
2. Teknik ini mampu menyediakan banyak sekali informasi dari hanya jumlah kecil data.

Kelemahan dari teknik *Forward Chaining* adalah sebagai berikut :

1. Kemungkinan tidak adanya cara untuk mengenali dimana beberapa fakta lebih penting dari fakta lainnya.
2. Sistem bisa saja menanyakan pertanyaan yang tidak berhubungan. Walaupun jawaban dari pertanyaan tersebut penting. Namun hal ini akan membingungkan user untuk menjawab pada subjek yang tidak berhubungan.

Untuk memahami cara kerja *Forward Chaining*, perhatikan contoh berikut ini :

Misalkan diketahui sistem pakar menggunakan 5 buah Rule berikut .

R1 : IF (Y AND D) THEN Z

R2 : IF (X AND B AND E) THEN Y

R3 : IF A THEN X

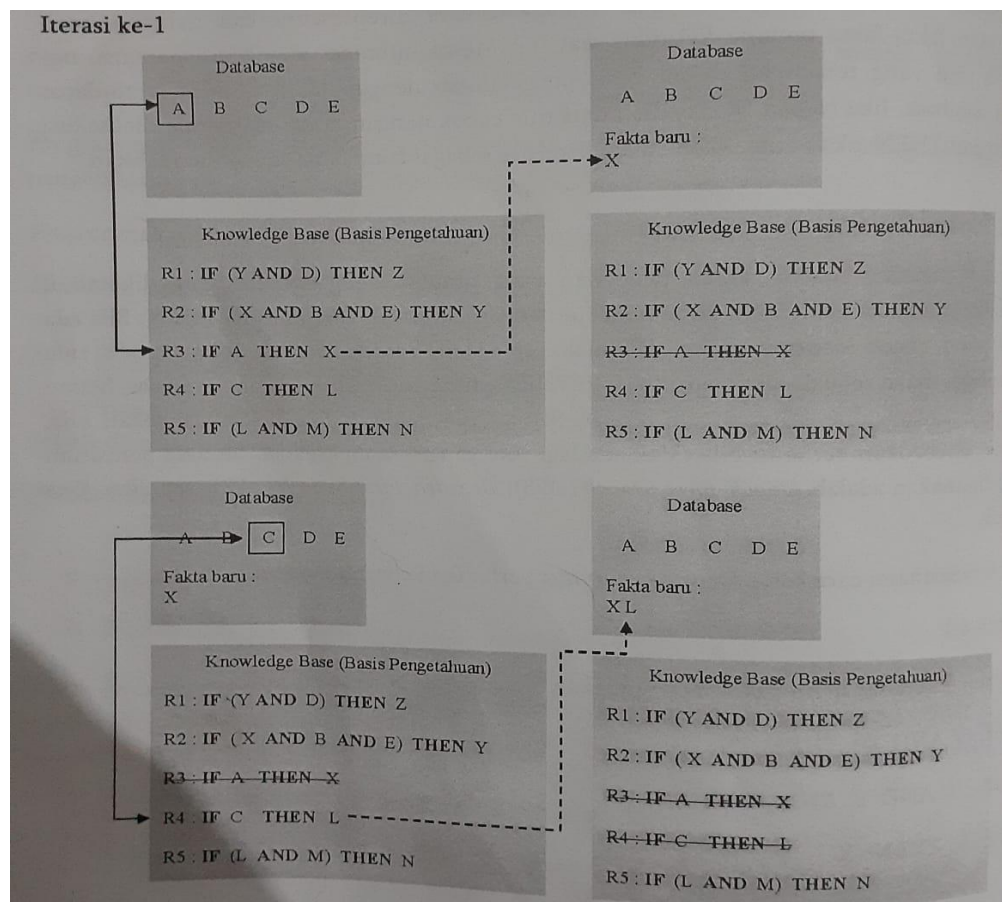
R4 : IF C THEN L

R5 : (L AND M) THEN N

Fakta-fakta : A,B,C,D,E dan E bernilai benar

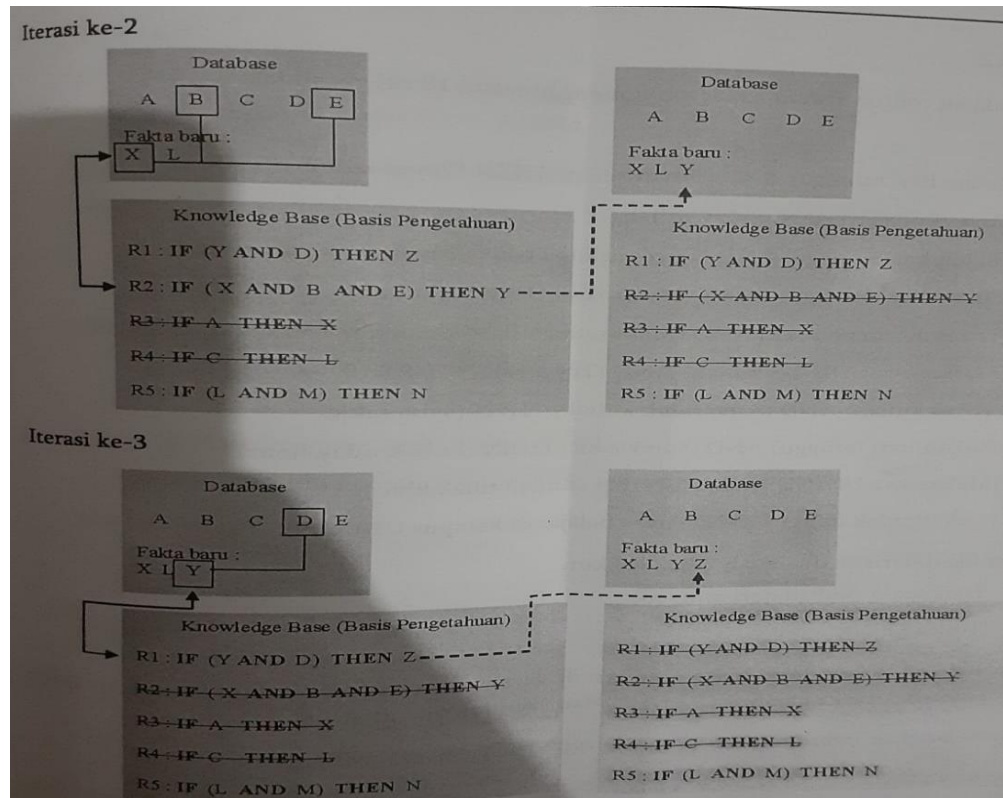
Goal : menentukan apakah Z bernilai benar atau salah

Gambar 2.3 berikut ini menguraikan diagram pelacakan kedepan (*Forward Chainig*) :



Gambar 2.3 Proses *Forward Chaining*

Gambar 2.4 berikut ini menguraikan diagram pelacakan kedepan (*Forward Chainig*) :



Gambar 2.4 Proses *Forward Chaining*

Sampai di sini proses dihentikan karena sudah tidak ada lagi rule yang bisa dieksekusi. Hasil pencarian adalah Z bernilai benar (lihat database di bagian fakta baru).

2.3.2 Dempster-Shafer Theory

Kusuma dewi (2003:100) menguraikan bahwa dalam menghadapi suatu permasalahan sering ditemukan jawaban yang tidak memiliki kepastian penuh. Ketidakpastian ini dapat berupa probabilitas atau kebolehjadian yang tergantung dari hasil suatu kejadian Hasil yang tidak pasti disebabkan oleh dua faktor, yaitu aturan yang tidak pasti dan jawaban pengguna yang tidak pasti atas suatu pertanyaan yang diajukan oleh sistem. Hal ini sangat mudah dilihat pada sistem diagnosis penyakit, dimana pakar tidak dapat

mendefinisikan hubungan antara gejala dengan penyebabnya secara pasti, dan pasien tidak dapat merasakan suatu gejala dengan pasti pula. Akhirnya akan ditemukan banyak kemungkinan diagnosis.

Sistem pakar harus mampu bekerja dalam ketidakpastian. Sejumlah teori telah ditemukan untuk menyelesaikan ketidakpastian, termasuk diantaranya probabilitas klasik, probabilitas Bayes, teori Hartley berdasarkan himpunan klasik, teori Shannon berdasarkan pada probabilitas, teori *Dempster-Shafer*, teori Fuzzy Zadeh, dan Faktor Kepastian. Untuk mengatasi ketidakpastian tersebut maka digunakan penalaran statistik, yang salah satunya menggunakan penalaran dengan teori Dempster-Shafer. Teori Dempster-Shafer adalah suatu teori matematika untuk pembuktian berdasarkan belief functions and plausible reasoning (fungsi kepercayaan dan pemikiran yang masuk akal), yang digunakan untuk mengkombinasikan potongan informasi yang terpisah (bukti) untuk mengkalkulasi kemungkinan dari suatu peristiwa. Teori ini dikembangkan oleh Arthur P. Dempster dan Glenn Shafer.

Secara umum teori Dempster-Shafer ditulis dalam suatu interval:

$$[Belief, Plausability] \dots \dots \dots (1)$$

Belief (*Bel*) adalah ukuran kekuatan evidence dalam mendukung suatu himpunan proposisi. Jika bernilai 0 maka mengindikasikan bahwa tidak ada *evidence*, dan jika bernilai 1 menunjukkan adanya kepastian. *Plausibility* (*Pls*) akan mengurangi tingkat kepastian dari *evidence*. *Plausibility* bernilai 0 sampai 1. Jika yakin akan X' , maka dapat dikatakan bahwa $Bel(X') = 1$, sehingga rumus di atas nilai dari $Pls(X) = 0$. Menurut Giarratano dan Riley fungsi *Belief* dapat diformulasikan sebagai:

$$Bel(X) = \sum_{Y \subseteq X} m(Y) \dots \dots \dots (2)$$

Sedangkan *plausibility* dinotasikan sebagai :

$$Pls(X) = 1 - Bel(X) = 1 - \sum_{Y \subseteq X} m(Y) \dots\dots\dots(3)$$

Dimana :

$Bel(X) = Belief(X)$

$Pls(X) = Plausibility(X)$

$m(X) = Mass\ function\ dari\ (X)$

$m(Y) = Mass\ function\ dari\ (Y)$

Teori *Dempster-Shafer* menyatakan adanya *Frame of discrement* yang dinotasikan dengan simbol (Θ). *Frame of discrement* merupakan semesta pembicaraan dari sekumpulan hipotesis sehingga sering disebut dengan *environment* :

$$\Theta = \{\theta_1, \theta_2, \dots, \theta_N\} \dots\dots\dots(4)$$

Dimana :

$\Theta = Frame\ of\ discrement\ atau\ environment$

$\theta_1, \dots, \theta_N = Elemen / unsur\ bagian\ dalam\ environment$

Environment mengandung $\rightarrow$ elemen-elemen yang menggambarkan kemungkinan sebagai jawaban, dan hanya ada satu yang akan sesuai dengan jawaban yang dibutuhkan. Kemungkinan ini dalam teori *Dempster-Shafer* disebut dengan *powerset* dan dinotasikan dengan $P(\Theta)$, setiap elemen dalam power set ini memiliki nilai interval antara 0 sampai 1, untuk $m:P(\Theta) \quad [0,1]$ sehingga dapat dirumuskan :

$$\sum_{X \in P(\Theta)} m(X) = 1 \dots\dots\dots(5)$$

Dengan :

$P(\Theta) = Power\ set$

$m(X) = Mass\ function\ (X)$

Mass function (m) dalam teori *Dempster-shafer* adalah tingkat kepercayaan dari suatu *evidence* (gejala), sering disebut dengan *evidence measure* sehingga dinotasikan dengan (m). Tujuannya adalah mengaitkan ukuran kepercayaan elemen-elemen θ . Tidak semua *evidence* secara langsung mendukung tiap-tiap elemen. Untuk itu perlu adanya probabilitas fungsi densitas (m). Nilai m tidak hanya mendefinisikan elemen-elemen θ saja, namun juga semua subsetnya. Sehingga jika θ berisi n elemen, maka subset θ adalah 2^n . Jumlah semua m dalam subset θ sama dengan 1. Apabila tidak ada informasi apapun untuk memilih hipotesis, maka nilai : $m\{\theta\} = 1,0$.

Apabila diketahui X adalah subset dari θ , dengan m_1 sebagai fungsi densitasnya, dan Y juga merupakan subset dari θ dengan m_2 sebagai fungsi densitasnya, maka dapat dibentuk fungsi kombinasi m_1 dan m_2 sebagai m_3 , yaitu:

$$m_3(Z) = \frac{\sum_{X \cap Y = Z} m_1(X).m_2(Y)}{\sum_{X \cap Y = \emptyset} m_1(X).m_2(Y)} \dots\dots\dots(6)$$

Dimana :

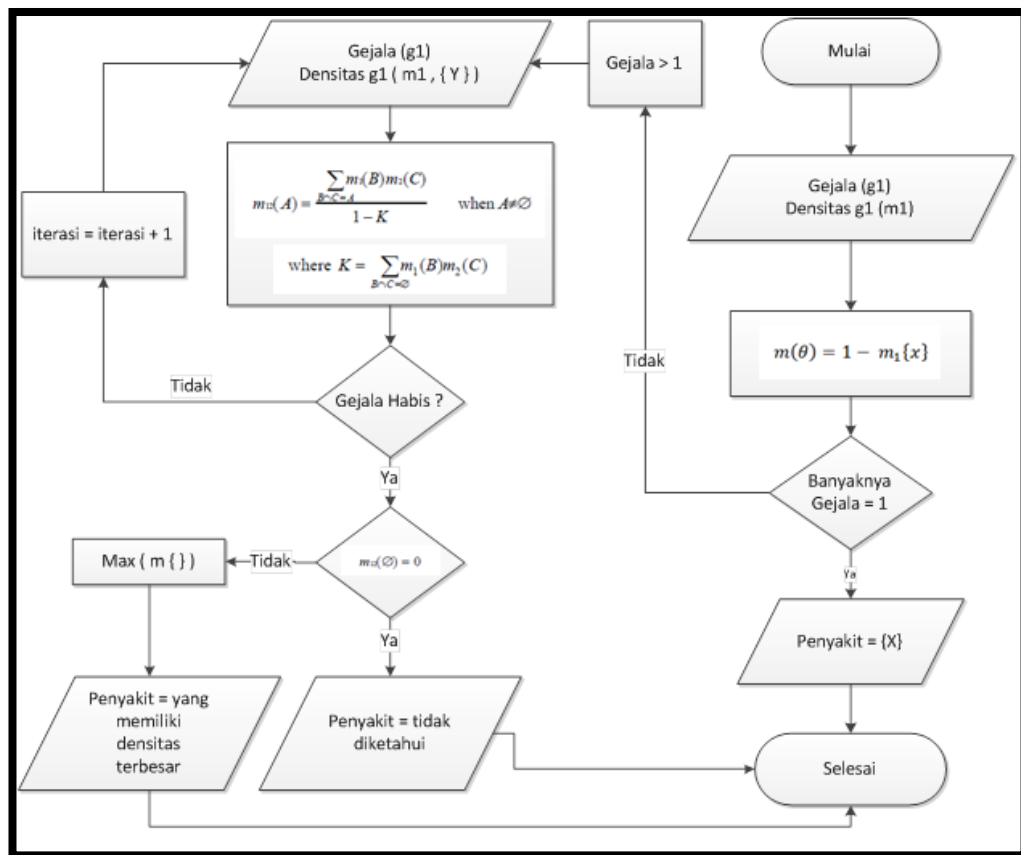
$m_3(Z)$ = *Mass function* dari *evidence* (Z)

$m_1(X)$ = *Mass function* dari *evidence* (X), yang diperoleh dari nilai keyakinan suatu *evidence* dikalikan dengan nilai *disbelief* dari *evidence* tersebut,

$m_2(Y)$ = *Mass function* dari *evidence* (Y), yang diperoleh dari nilai keyakinan suatu *evidence* dikalikan dengan nilai *disbelief* dari *evidence* tersebut.

$\sum m_1(X).m_2(Y)$ = Merupakan nilai kekuatan dari *evidence* Z yang diperoleh dari kombinasi nilai keyakinan sekumpulan *evidence*.,

Gambar 2.5 berikut ini merupakan gambaran algoritma *Dempster-Shafer*, dalam memudahkan pembacaan algoritma DST:



Gambar 2.5 Algoritma Dempster-Shafer

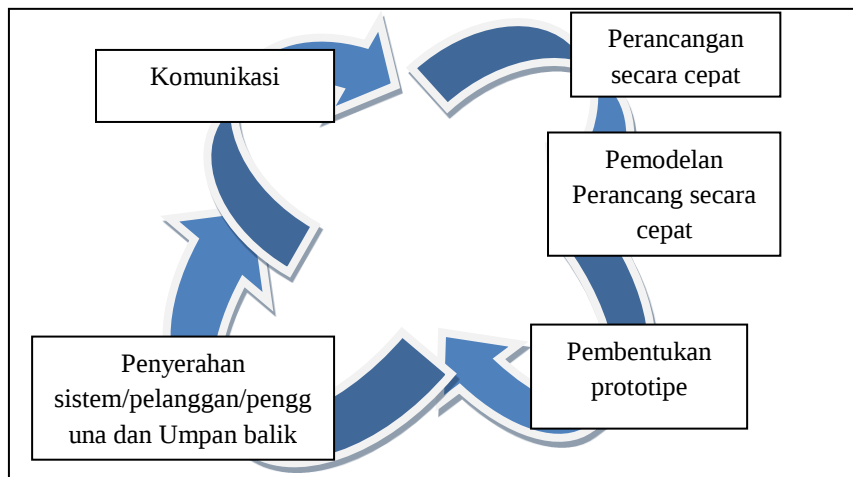
2.4 Metode Pengembangan Perangkat Lunak

Metode perancangan sistem yang digunakan dalam penelitian ini adalah dengan menggunakan model proses perancangan perangkat lunak Prototype melalui paradigm atau pendekatan berorientasi objek yang dimodelkan menggunakan *Unified Modeling Language* (UML).

2.4.1 Pembuatan Prototipe Atau Metode (*Prototyping*)

Roger S. Pressman, Ph. D. (2012 : 51) Menguraikan bahwa seringkali pelanggan mendefinisikan sejumlah sasaran perangkat lunak secara umum, tetapi tidak bisa mengidentifikasi spesifikasi kebutuhan yang rinci untuk fungsi-fungsi dan fitur-fitur yang nantinya akan dimiliki perangkat lunak yang akan dikembangkan. Dalam kasus yang lain, pengembangan perangkat

lunak mungkin merasa tidak pasti tentang efesiensi suatu algoritma yang akan digunakan dalam pengembangan perangkat lunak, atau juga merasa tidak pasti akan kemampuan perangkat lunak untuk beradaptasi dengan sistem operasi yang akan digunakan. Gambar 2.6 berikut ini menguraikan metode *Prototype* :



Gambar 2.6 Metode *Prototype* (Pressman, 2012: 51)

Tahapan-tahapan dalam metode *prototye* adalah sebagai berikut :

- 1.) Komunikasi dan pengumpulan data awal, yaitu analisis terhadap kebutuhan pengguna.
- 2.) Perancangan secara cepat (desain cepat), yaitu pembuatan desain secara umum untuk selanjutnya dikembangkan kembali
- 3.) Pemodelan perancangan secara cepat yaitu memproduksi perangkat prototype termasuk pengujian dan penyempurnaan
- 4.) Pembuatan *prototype* aplikasi yaitu mengevaluasi *prototype* dan memperhalus analisis terhadap kebutuhan pengguna
- 5.) Penyerahan sistem/Perangkat lunak ke para pelanggan/ pengguna yaitu pembuatan tipe yang sebenarnya berdasarkan hasil dari evaluasi *prototype*

Kelebihan dari metode prototyping ini adalah sebagai berikut :

1. Adanya komunikasi yang baik antara pengembang dan pelanggan
2. Pengembangan dapat bekerja baik dalam menentukan kebutuhan pelanggan
3. Lebih menghemat waktu dalam pengembangan sistem
4. Penerapan lebih mudah karena pemakai mengetahui apa yang diharapkannya

Kekurangan dari metode prototyping ini adalah sebagai berikut :

1. Resiko tinggi yaitu untuk masalah - masalah yang tidak terstruktur dengan baik, ada perubahan yang besar dari waktu ke waktu dan adanya persyaratan data yang tidak menentu
2. Interaksi pemakai penting . Sistem harus menyediakan dialog *on-line* antara pelanggan dan komputer
3. Hubungan pelanggan dengan komputer yang disediakan mungkin tidak mencerminkan teknik perancangan yang baik.

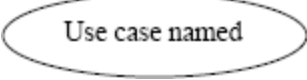
2.4.2 *Unified Modeling Language (UML)*

Rosa A. S. Dan M. Shalahuddin (2019:133) menguraikan bahwa *Unified Modeling Language (UML)* Adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requitremment*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek.

1.) *Use Case Diagram*

Rosa A. S. Dan M. Shalahuddin (2019:155) *Use case* atau diagram *use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. Tabel 2.1 berikut ini menguraikan simbol-simbol *use case* :

Tabel 2.1 Simbol-Simbol Use Case.

SIMBOL	NAMA	KETERANGAN
	Actor	Actor adalah pengguna sistem. Actor tidak terbatas hanya manusia saja, jika sebuah sistem berkomunikasi dengan aplikasi lain dan membutuhkan input atau memberikan output, maka aplikasi tersebut juga bisa dianggap sebagai actor.
	Use Case	Use case digambarkan sebagai lingkaran elips dengan nama use case dituliskan didalam elips tersebut.
	Association	Asosiasi digunakan untuk menghubungkan actor dengan use case. Asosiasi digambarkan dengan sebuah garis yang menghubungkan antara Actor dengan Use Case.

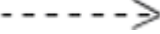
2.) Class Diagram

Rosa A. S. Dan M. Shalahuddin (2019:141) menguraikan bahwa diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendenifisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki 3 area pokok yaitu :

- Nama, kelas harus mempunyai sebuah nama.
- Atribut, adalah kelengkapan yang melekat pada kelas. Nilai dari suatu kelas hanya bisa diproses sebatas atribut yang di miliki.
- Operasi, adalah proses yang dapat dilakukan oleh sebuah kelas, baik pada kelas itu sendiri ataupun kepada kelas lainnya. Tabel 2.2 berikut ini menguraikan Simbol class diagram :

Tabel 2.2 Simbol Class Diagram

GAMBAR	NAMA	KETERANGAN
--------	------	------------

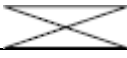
	<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
	<i>N-Ary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
	<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
	<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor.
	<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
	<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
	<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.

3.) Activity Diagram

Rosa A. S. Dan M. Shalahuddin (2019:161) menguraikan bahwa diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dilakukan oleh sistem. Tabel 2.3 berikut ini menguraikan Simbol Activity Diagram :

Tabel 2.3 Simbol Activity Diagram

SIMBOL	KETERANGAN
--------	------------

	Titik Awal
	Titik Akhir
	Activity
	Pilihan Untuk mengambil Keputusan
	Fork; Digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau untuk menggabungkan dua kegiatan paralel menjadi satu.
	Rake; Menunjukkan adanya dekomposisi
	Tanda Waktu
	Tanda pengiriman
	Tanda penerimaan
	Aliran akhir (Flow Final)

2.5 Perangkat Lunak Pengembangan Sistem

2.5.1 Web Server

MADCOMS (2016:4) menguraikan bahwa *web server* adalah suatu program komputer yang mempunyai tanggung jawab atau tugas menerima permintaan HTTP dari komputer klien, yang dikenal dengan nama web browser dan melayani mereka dengan menyediakan respon HTTP berupa konten data, biasanya berupa halaman web yang terdiri dari dokumen HTML dan objek terkait seperti gambar dan lain-lain.

2.5.2 MySQL (*database*)

MADCOMS (2016:2) menguraikan bahwa *mySQL* adalah sistem manajemen database SQL yang bersifat Open Source dan paling populer saat ini. Sistem database *mySQL* mendukung beberapa fitur seperti multithreaded, multi-user, dan SQL database management system (DBMS). Database ini dibuat untuk keperluan sistem database yang cepat, handal, dan mudah digunakan.

2.5.3. Bahasa Pemrograman

2.5.3.1 HTML (*Hypertext Markup Language*)

Sutarman (2007:7). Menguraikan bahwa Pengertian HTML (*Hyper Text Mark-up Language*) adalah suatu bahasa yang digunakan untuk menulis halaman web. HTML dirancang untuk digunakan tanpa tergantung pada suatu platform tertentu platform independent).

2.5.3.2 PHP (*Personal Home Pages*)

MADCOMS (2016:2) menguraikan bahwa PHP adalah bahasa script yang dapat ditanamkan atau disisipkan ke dalam HTML. PHP banyak dipakai untuk membuat program situs web dinamis. PHP sering juga digunakan untuk membangun sebuah CMS.

2.5.3.3 CSS (*Cascading Style Sheets*)

Shalahuddin dan Sukamto (2010:45) menguraikan bahwa “CSS atau Cascading StyleSheets adalah suatu fasilitas untuk mempermudah pemeliharaan sebuah halaman web, dengan menggunakan CSS sebuah halaman web dapat diubah tampilannya tanpa harus mengubah dokumen HTML-nya. CSS dapat ditulis dengan huruf besar maupun huruf kecil seperti halnya HTML karena CSS non-case sensitive (tidak membedakan huruf besar dan huruf kecil).

2.5.3.4 JavaScript

Hidayatullah dan kawistara (2015: 422) menguraikan bahwa “Java Script adalah bahasa skrip (Scripting Language) yang digunakan

sebagai fungsionalitas dalam membuat suatu web, Javascript sendiri bersifat client-side sehingga untuk menggunakannya browser anda harus mengaktifkan fitur javascript (bila erkonfigurasi browser anda diset pada konfigurasi default, secara otomatis browser anda sudah mengaktifkan fitur javascript”.

2.5.4 Hosting

Sugiyanto (2013:55) menguraikan bahwa Hosting berasal dari kata host. Komputer yang terhubung dalam jaringan. Memanfaatkan fasilitas yang tersedia dalam suatu computer yang terhubung dengan jaringan. Hosting menyediakan sumber daya server-server untuk disewakan sehingga memungkinkan organisasi atau individu menmpatkan informasi di internet. Server hosting terdiri dari gabungan server-server atau sebuah server yang terhubung dengan jaringan internet berkecepatan tinggi.

2.6 Pengujian Perangkat Lunak

Rosa A. S. Dan M. Shalahuddin (2019:271) menguraikan bahwa sebuah perangkat lunak perlu dijaga kualitasnya bahwa kualitas bergantung kepada kepuasan pelanggan (*customer*). Kualitas perangkat lunak perlu dijaga untuk keperluan sebagai berikut :

1. Agar dapat “survive” bertahan hidup di dunia bisnis perangkat lunak
2. Dapat bersaing dengan perangkat lunak lain.
3. Penting untuk pemasaran global (*global marketing*).

2.6.1 Black Box Testing

Rosa A. S. Dan M. Shalahuddin (2019:275) menguraikan bahwa menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spsifikasi yang dibutuhkan. Pengujian kotak hitam dilakukan dengan membuat kasus uji yang bersifat mencoba semua fungsi dengan memakai perangkat lunak

apakah sesuai dengan spesifikasi yang dibutuhkan. Kasus uji yang dibuat untuk melakukan pengujian kotak hitam harus dibuat dengan kasus benar dan kasus salah, misalkan untuk kasus proses login maka kasus uji yang dibuat adalah :

1. Jika user memasukkan nama pemakai (*username*) dan kata sandi (*password*) yang benar
2. Jika user memasukan nama pemakai (*username*) dan kata sandi (*password*) yang salah, misalnya nama pemakai benar tapi kata sandi salah, atau sebaliknya, atau keduanya salah.

2.7 Pengertian diagnosis

Diagnosis sering kita dengar dalam istilah medis. Menurut Thorndike dan Hagen dalam Suherman (2011:5) diagnosis dapat diartikan sebagai berikut:

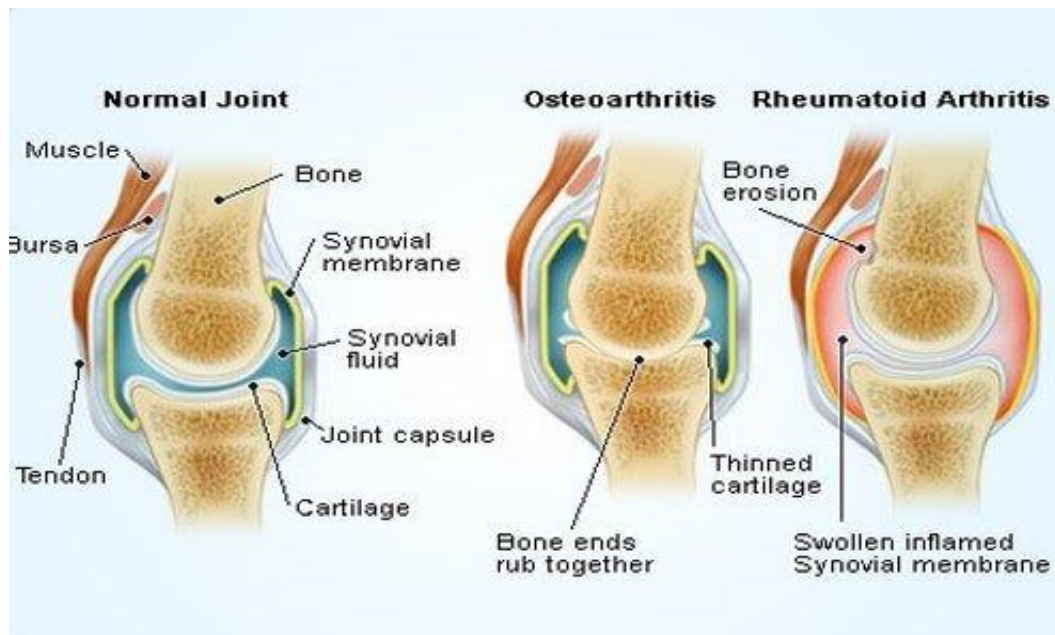
- 1.) Upaya atau proses menemukan kelemahan atau penyakit (*weakness, disease*) apa yang dialami seseorang dengan melalui pengujian dan studi yang seksama mengenai gejala-gejalanya (*symptoms*).
- 2.) Studi yang seksama terhadap fakta tentang suatu hal untuk menemukan karakteristik atau kesalahan-kesalahan dan sebagainya yang esensial.
- 3.) Keputusan yang dicapai setelah dilakukan suatu studi yang seksama atas gejala-gejala atau fakta tentang suatu hal.

2.8 Sendi (Artikulasi)

Prieharti dan dr. Yekti Mumpuni (2017: 11) menguraikan bahwa sendi merupakan bagian tubuh yang berfungsi sebagai penghubung tulang yang satu dengan yang lain. Dengan adanya sendi, tulang-tulang dapat digerakan. Sendi bekerja seperti engsel. Dalam melakukan tugasnya, sendi melibatkan / membutuhkan otot, tendon, ligamen, dan tulang rawan. Ada berbagai macam penyakit sendi pada manusia, beberapa penyakit tersebut antara lain: *Gout, rematik arthritis (RA), spondilitis ankilosa, lupus dan osteoarthritis (OA)*.

2.8.1 Arthritis

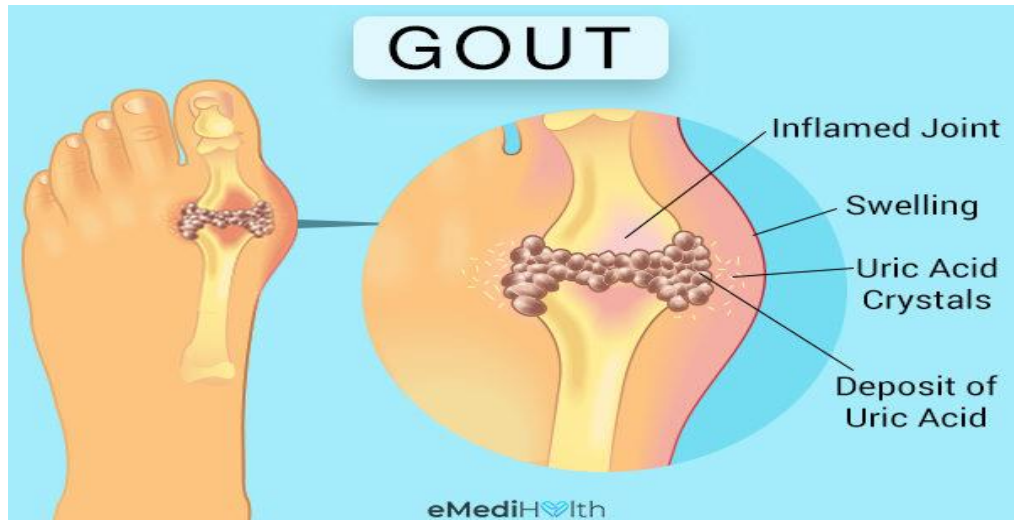
Prieharti dan dr. Yekti Mumpuni (2017: 15) menguraikan bahwa istilah medis peradangan pada sendi. *Arthritis* berasal dari bahasa Yunani, *arthros* (sendi) dan *itis* (inflamasi). Inflamasi adalah reaksi tubuh terhadap benda asing yang ditandai oleh panas, bengkak, nyeri dan gangguan fungsi organ tubuh. Gambar 2.7 di bawah ini merupakan penderita radang sendi :



Gambar 2.7 Penderita Radang Sendi

2.8.1.1 Gout Arthritis

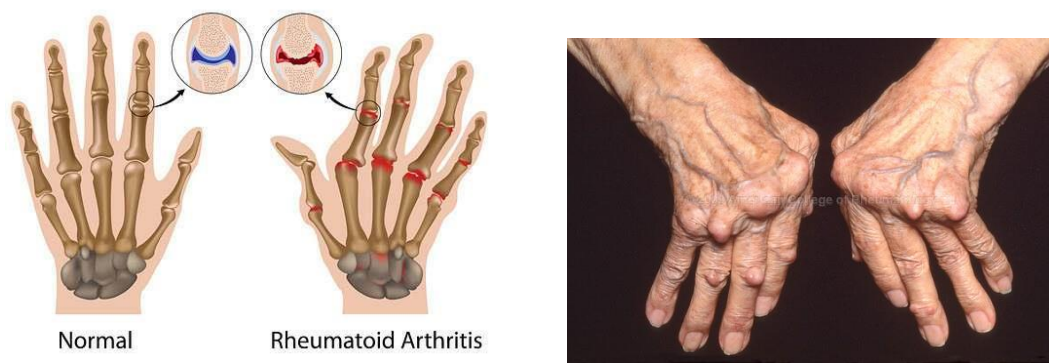
Prieharti dan dr. Yekti Mumpuni (2017: 19) menguraikan bahwa gout atau penyakit asam urat diakibatkan gangguan metabolisme asam urat dalam tubuh. Gambar 2.8 di bawah ini merupakan penderita penyakit Gout / Asam urat:



Gambar 2.8 Penyakit Gout / Asam urat

2.8.1.2 Rematik arthritis

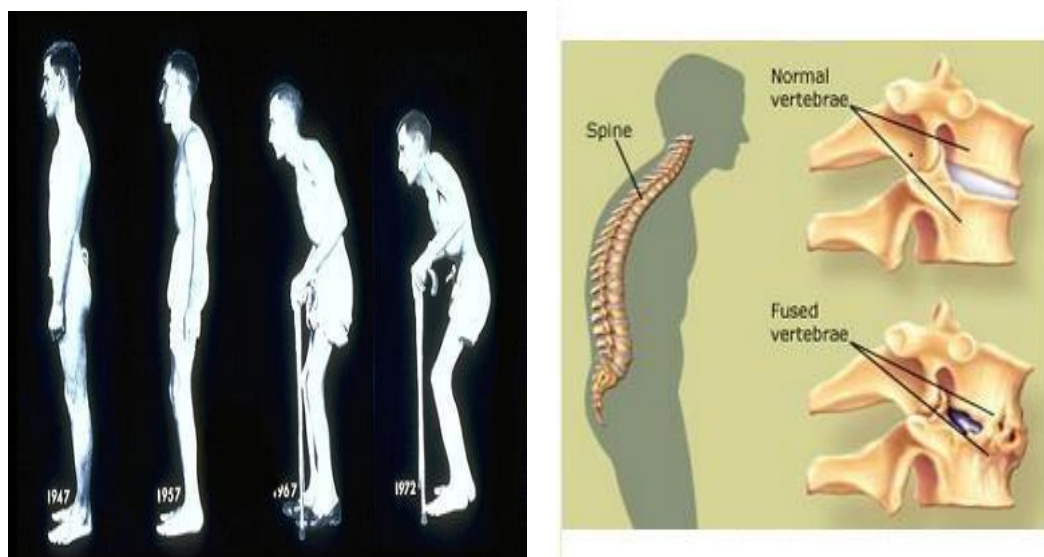
Prieharti dan dr. Yekti Mumpuni (2017: 12) menguraikan bahwa Rematik arthritis merupakan rematik yang disebabkan karena gangguan sistem kekebalan tubuh yang memengaruhi sendi dan jaringan lain. Gambar 2.9 di bawah ini merupakan penderita penyakit *Rematik Arthritis* (SA) :



Gambar 2.9 Penderita *Rematik Arthritis*

2.8.1.3 Spondilitis Ankilosa (SA)

Prieharti dan dr. Yekti Mumpuni (2017: 23-25) menguraikan bahwa *spondilitis ankilosa* (SA) berasal dari kata *angkylosing* (tulang yang bergabung bersama sepanjang sendi) dan *spondylitis* (inflamasi tulang belakang), SA merupakan penyakit peradangan yang berawal dari sendi panggul dan menyebar ke tulang belakang. Gambar 2.10 berikut ini merupakan penderita penyakit *Spondilitis Ankilosa* (SA).



Gambar 2.10 Penderita *Spondilitis Ankilosa*

2.8.1.4 Lupus

Lupus merupakan jenis arthritis di mana sistem imun menyerang tubuh. Nama ini didapatkan dari tanda umum khas penderita lupus, yaitu ruam mirip kupu-kupu (Mumpuni 2017: 26-28). Gambar 2.11 di bawah ini merupakan penderita penyakit Lupus :



Gambar 2.11 Penderita Lupus

2.8.1.5 Osteoarthritis (OA)

Osteoarthritis selanjutnya disebut OA, merupakan jenis penyakit sendi akibat proses degeneratif sekaligus peradangan (inflamasi) pada tulang rawan sendi. Penyakit sendi degeneratif adalah kemunduran (perubahan menjadi sesuatu yang rusak) bertahap kartilago artikular pada sendi, disertainya dengan perubahan jaringan lunak di sekitar sendi (Mumpuni 2017: 29). Berikut ini merupakan jenis-jenis osteoarthritis adalah sebagai berikut:

- 1.) OA primer, terjadi pada laki-laki usia pertengahan dan pada wanita usia lebih tua (dialami setelah usia 45 tahun), menyerang secara perlahan tapi progresif, dan dapat mengenai lebih dari satu persendian. Biasanya menyerang sendi pada berat badan seperti lutut, panggul, menyerang punggung, leher, serta jari-jari.

2.) OA sekunder, terjadi pada setiap usia dan abnormal sejak lahir (dialami sebelum usia 45 tahun), biasanya disebabkan karena trauma yang menyebabkan luka pada sendi seperti patah tulang atau permukaan sendi tidak sejajar, akibat sendi yang longgar, dan pembedahan pada sendi. Perubahan yang paling mencolok dari osteoarthritis biasanya terdapat didaerah tulang rawan yang mendapat beban pada stadium awal, tulang rawan lebih tebal daripada normal, tetapi seiring dengan perkembangan OA permukaan sendi menipis, tulang rawan melunak, integritas permukaan terputus dan terbentuk celah vertikal (Mumpuni 2017: 30). Gambar 2.12 berikut ini merupakan penderita penyakit *Osteoarthritis* (OA) :



Gambar 2.12 Penderita Penyakit *Osteoarthritis*

2.9 Penelitian Terkait

Penelitian ini membandingkan berbagai referensi dari beberapa jurnal tentang pembahasan mengenai permasalahan yang sama dihadapi dengan peneliti. Tabel 2.5 yang ada dibawah ini adalah tabel penelitian terkait.

Tabel 2.5 Penelitian Terkait

No	Nama	Terbit / Tahun	Judul	Uraian
1.	Mikha Dayan Sinaga, Nita Sari Br. Sembiring	mikha_dayan@yahoo.co.id tata_olala@yahoo.co.id (2016)	Penerapan Metode Dempster Shafer Untuk Mendiagnosa Penyakit Dari Akibat Bakteri Salmonella	Salah satu bakteri yang dapat menimbulkan penyakit adalah bakteri Salmonella. Salmonella adalah bakteri gram negatif, berbentuk spora yang memfermentasi glukosa menjadi Enterobacteria. Salmonella dapat menyebabkan berbagai macam penyakit. Salmonella menyerang saluran yang mencakup perut, usus halus, usus besar atau kolon Metode yang digunakan dalam penelitian ini adalah metode Dempster-Shafer Variable yang digunakan dalam penelitian ini adalah penyakit, jenisnya ,Gejala dan tipe penyakit Aplikasi ini hanya memberikan informasi diagnosa penyakit dari Bakteri Salmonella., berbeda dengan penelitian ini yaitu memberikan informasi tentang diagnosa penyakit dari Bakteri Salmonella tanpa penelitian dilaboratorium sehingga hasilnya kurang tepat. Kelebihan yang ada pada aplikasi saya yaitu, menggunakan metode 2 metode, metode yang pertama adalah metode penalaran (Forward Chaining) dan metode ketidakpastian (dempster-shafafer) sehingga menghasilkan perhitungan yang akurat.
2.	Henny Destiana, Fajar Sidik Andhika	henny.hnd@bsi.ac.id , fajarsidikandhika@gmail.com (2017)	PERANCANGAN APLIKASI SISTEM PAKAR UNTUKMENDIAGNOSA PENYAKIT GIGI DENGAN	Kesehatan gigi dan mulut merupakan hal yang sangat penting bagi setiap manusia, karena kesehatan gigi dan mulut dapat mempengaruhi kesehatan tubuh secara menyeluruh. mulut merupakan suatu tempat yang sangat ideal bagi

			METODE DEMSPTER-SHAFER perkembangbiakan bakteri karena temperatur, kelembaban, dan makanan cukup tersedia disana. juga karena pada mulut terdapat beberapa fisur gigi, sehingga sisa makanan mudah tertinggal, hal ini merupakan makanan yang amat disukai oleh bakteri. Metode yang digunakan dalam penelitian ini adalah metode Dempster-Shafer Variable yang digunakan dalam penelitian ini adalah penyakit, Gejala, tipe penyakit. Sistem ini hanya memberikan informasi pendeteksi resiko <i>Osteoporosis</i> dan <i>osteoarthritis</i> , berbeda dengan penelitian ini yang memberikan informasi tentang satu kepakaran untuk satu jenis penyakit. Kelebihan pada aplikasi saya yaitu, memuat lebih banyak variable antara lain : Diagnosa penyakit, jenis penyakit, pola hidup sehat, tentang website, profil, dan riwayat diagnosa.	
3.	Reza Amelia, Ause Labellapansa 2, Apri Siswanto	1rezaamelia94@gmail.com, 2a.use.labella@eng.uir.ac.id, 3aprisiswanto@eng.uir.ac.id (2018)	Sistem Pakar Sebagai Alat Bantu Untuk Pendekatan Diagnosis Penyakit Thalasemia Pada Anak Menggunakan Metode Dempster-Shafer	Adapun salah satu penyakit genetik adalah Thalasemia. Thalasemia adalah salah satu jenis penyakit kelainan darah bawaan. Penyakit ini biasanya ditandai dengan dengan kondisi sel darah merah (eritrosit) yang mudah rusak atau lebih pendek umurnya dari sel darah normal pada umumnya, yaitu 120 hari. Kondisi ini diturunkan orang tua kepada anaknya sejak dalam kandungan. Saat ini masalah penyakit Thalasemia tetap banyak, bahkan akan terus meningkat. Disamping itu penyebab peningkatan kasus penyakit Thalasemia pada anak ini disebabkan rendahnya kesadaran masyarakat untuk melakukan pemeriksaan dan bagaimana cara pengobatan untuk penyakit Thalasemia tersebut. Pada zaman modern sekarang ada sebuah aplikasi yang mempermudah kerja pakar untuk membantu mendiagnosis penyakit Thalasemia yaitu sistem pakar.

				Metode yang digunakan dalam penelitian ini adalah metode Dempster-Shafer. Variable yang digunakan dalam penelitian ini adalah penyakit Gejala dan tipe penyakit. Aplikasi ini hanya memberikan informasi jenis gejala penyakit, tanpa memberi informasi pertanyaan kepasien. Kelebihan yang ada pada aplikasi saya yaitu, menggunakan metode 2 metode, metode yang pertama adalah metode penalaran (Forward Chaining) dan metode ketidakpastian (dempster-shafafer) sehingga menghasilkan perhitungan yang akurat.
4.	Septilia Arfida	septiliatime@gmail.com (2012)	IMPLEMENTASI FUZZY TERHADAP SISTEM PENDUKUNG KEPUTUSAN DALAM MENENTUKAN KELAYAKAN PEMBIAYAAN PENGAJUAN KREDIT BARANG	Peranan teknologi informasi dan komputer adalah dalam mendukung pengolahan data agar lebih efektif dan efisien. Banyak perusahaan swasta maupun instansi pemerintah yang telah memanfaatkan teknologi komputer guna membantu mengolah data sesuai dengan kebutuhannya dan sistem pendukung keputusan adalah salah satu dari pemanfaatan teknologi komputer yang dapat membantu dalam pengambilan suatu keputusan. Metode yang digunakan dalam penelitian ini adalah metode Mamdani Variable yang digunakan dalam penelitian ini adalah penghasilan, pengajuan, dan administrasi sebagai variable input dan outputnya adalah keputusan. Sistem ini hanya memberikan informasi keputusan kelayakan dan kemampuan konsumen dalam pengajuan kredit barang.

				Kelebihan yang ada pada aplikasi saya yaitu, menggunakan metode 2 metode, metode yang pertama adalah metode penalaran (Forward Chaining) dan metode ketidakpastian (dempster-shafer) sehingga menghasilkan perhitungan yang akurat. serta variable yang lebih banyak seperti, Diagnosa penyakit, jenis penyakit, pola hidup sehat, tentang website, profil, dan riwayat diagnosa.
5.	Septilia Arfida, Mohammad Sholeh2	septilia@darmaia.ac.id muh_sholeh@yuh.sholeh@mail.com (2019)	Sistem Pendiagnosa Penyakit Hipokalemia Menggunakan Metode Fuzzy Inference System Tsukamoto Berbasis Android	Teknologi informasi mengalami perkembangan yang sangat pesat. Salah satu perkembangan teknologi yaitu teknologi smart phone Android. Smart phone dapat dimanfaatkan dalam berbagai bidang salah satunya bidang kesehatan. Kesehatan erat kaitannya dengan penyakit yang diderita pada manusia. Salah satu penyakit yang berbahaya namun sering masyarakat tidak mengetahuinya adalah Hipokalemia. Hipokalemia adalah kekurangan zat kalium dalam darah manusia. Kalium itu sendiri adalah mineral yang paling penting dan mengandung ion bermuatan positif dalam sel-sel tubuh. Kalium menjaga fungsi jantung, otak, ginjal, jaringan otot, dan organ tubuh lainnya agar selalu dalam kondisi sehat. Metode yang digunakan dalam penelitian ini adalah metode Fuzzy Tsukamoto. Variable yang digunakan dalam penelitian ini adalah jenis penyakit dan gejala penyakit. Sistem ini hanya memberikan informasi pendeteksi diagnosa persendian tanpa memberi informasi penanganan, perawatan, dan pencegahan penyakit hipokalemia. Kelebihan aplikasi saya yaitu sudah bersifat online, sedangkan peneliti sebelumnya masih bersifat offline.