

BAB II

LANDASAN TEORI

2.1 Jantung

2.1.1 Pengertian Jantung

Menurut dr. Djoko Maryono, 2008, Jantung adalah organ yang vital bagi manusia yang terletak di rongga dada sebelah kiri. Jantung dibagi menjadi 4 : 2 natrium (*atrium*/serambi kiri dan kanan) dan 2 *ventrikel*/bilik (*Ventrikel*/bilik kiri dan kanan). Pemompaan darah melalui keempat ruang tersebut dibantu oleh 4 katup jantung. Katup membuka dan menutup sehingga darah hanya mengalir dalam satu arah. Keempat katup jantung tersebut adalah :

- Katup *trikuspidalis*, terletak di antara serambi kanan dan bilik kanan
- Katup *pulmonalis*, terletak di antara bilik kanan dan arteri pulmonalis
- Katup *mitralis*, terletak di antara serambi kiri dan bilik kiri.
- Katup *aorta*, terletak di antara bilik kiri dan *aorta*. (dr. Djoko Maryono,2008)

2.1.2 Fungsi Jantung

Jantung berfungsi untuk memompa darah ke seluruh tubuh. Jantung bersifat sedemikian rupa canggihnya, kekuatan dan daya tahannya. Di sisi lain kesehatan jantung tergantung pada keseimbangan yang rentan. Bila bagian terkecil pun dari perangkat yang kompleks itu kurang mendapatkan perhatian, jantung dapat menyebabkan rasa yang tidak enak pada sekujur tubuh, pusing, sakit bahkan meninggal.

Disamping memiliki peranan fisik yang menentukan, jantung juga erat terkait dengan kadar emosi seseorang, karena jantung dianggap sebagai tempat berpangkalnya emosi. Bila terkena penyakit jantung dampaknya akan luas. Karena itu kecemasan dan ketegangan berpengaruh terhadap system kardiovaskular yang dapat tercermin pada detak jantung yang berdebar-debar, sesak nafas dan lain-lain. (Iman Soeharto, 2004)

2.1.3 Menjaga Kesehatan Jantung

Untuk menjaga aliran darah ke jantung cukup, rawatlah setiap bagian jantung dengan:

1. Menghindari hal-hal yang dapat merusak pembuluh jantung, misalnya kolesterol yang terlalu tinggi, gula darah yang terlalu tinggi. Semua itu dapat menyebabkan penyumbatan pembuluh darah jantung (pembuluh jantung koroner).
2. Hindari obat-obatan dan zat yang bisa merusak otot jantung, misalnya *kurare* (Semacam pelemas otot, seperti pada bisa ular), sianida dan alkohol.
3. Berkaitan dengan system penghantar araf jantung, saraf jantung bias rusak oleh virus-virus dan zat-zat tertentu seperti Pb (timah hitam).
4. Kuman tertentu dapat merusak katup jantung, seperti kuman *Beta streptococcus haemoliticus*. Kuman ini biasanya terdapat di tenggorokan. Jika tidak diberi antibiotik yang cukup.(dr. Djoko Maryono,2008)

2.1.4 Jenis-Jenis Penyakit Jantung

2.1.4.1 Penyakit Jantung Koroner (PJK)

Penyakit jantung koroner adalah penyempitan pembuluh darah kecil yang memasok darah dan oksigen ke jantung. Adapun gejala-gejala yang dapat dirasakan yaitu : nyeri dada secara tiba-tiba, keluarnya keringat dingin yang berlebihan, nyeri kepala berkepanjangan, seluruh tubuh terasa terbakar, mual dan muntah, tubuh cepat lelah dan lemas, mengalami sesak nafas, adanya pembengkakan di sekitar sendi dan kaki. Penyakit jantung koroner disebabkan oleh tekanan darah yang tinggi(hipertensi), diabetes, kolesterol, obesitas, merokok, mengkonsumsi alkohol dan faktor keturunan. (dr. Djoko Maryono,2008)

2.1.4.2 Penyakit Jantung Hipertensi (PJH)

Penyakit jantung hipertensi adalah suatu penyakit yang berkaitan dengan dampak sekunder pada jantung karena hipertensi sistemik yang berkepanjangan. Adapun gejala-gejala yang dapat dirasakan yaitu : sakit kepala, nyeri dada secara tiba-tiba,jantung berdebar-debar, tubuh cepat lelah dan lemas, mimisan. Penyakit jantung hipertensi disebabkan oleh tekanan darah yang tinggi(hipertensi), kolesterol, obesitas, stress, merokok, mengkonsumsi alkohol. Penyakit ini dapat dicegah dengan melakukan olahraga secara teratur, berhenti merokok dan mengkonsumsi alkohol, menghindari makanan berlemak dan berkolesterol tinggi, menjaga kesehatan mental, dan mengkonsumsi air putih minimal 1 setengah liter per hari. (dr. Djoko Maryono,2008)

2.1.4.3 Penyakit Jantung Perikarditis (PJP)

Penyakit jantung perikarditis adalah Peradangan pada kantung jantung atau perikardium sehingga menimbulkan penimbunan cairan dan penebalan. Adapun gejala-gejala yang dapat dirasakan yaitu : mual dan muntah, tubuh cepat lelah dan lemas, demam, sulit bernafas, batuk, dada terasa sakit, pembengkakan perut, sesak nafas. Penyakit jantung perikarditis disebabkan oleh Infeksi virus seperti *Coxsackie* dan *influenza*, Infeksi bakteri seperti *Streptococcus*, *Staphylococcus*, *Meningococcus*, dan *Gonococcus*, Infeksi parasit, Infeksi jamur., Menderita kanker, Menderita *infark miokard*, dan Menderita tuberkulosis. Penyakit ini dapat dicegah dengan anti biotik, anti jamur, *perikardiosentesis*, obat anti peradangan, *analgesik* dan aspirin. (dr. Djoko Maryono,2008)

2.1.4.4 Penyakit Jantung Rematik (PJR)

Penyakit jantung rematik adalah suatu kondisi dimana terjadi kerusakan pada katup jantung yang bisa berupa penyempitan, atau kebocoran, terutama katup mitral(stenosis katup mitral) sebagai akibat adanya gejala sisa dari Demam Rematik (DR). Adapun gejala-gejala yang dapat dirasakan yaitu : nyeri sendi yang berpindah-pindah, bercak kemerahan dikulit yang terbatas, gerakan tangan yang tak beraturan dan tak terkendali, sesak nafas, benjolan kecil-kecil dibawah kulit, nyeri perut, kehilangan berat badan, mudah lelah dan demam. Penyakit jantung rematik disebabkan oleh faktor genetik, umur, keadaan gizi, golongan etnik dan ras, jenis kelamin dan reaksi autoimun. Penyakit ini dapat dicegah dengan rajin membersihkan rumah, olahraga, mengkonsumsi makanan sehat, menghindari rokok, memakai masker di lingkungan berdebu. (dr. Djoko Maryono,2008)

2.1.4.5 Penyakit Otot Jantung

Penyakit otot jantung adalah hilangnya kemampuan jantung untuk memompa darah dan berdenyut secara normal. Adapun gejala-gejala yang dapat dirasakan yaitu : sesak nafas, pembengkakan kaki dan tangan, perut terasa kembung, mudah lelah, detak jantung tidak teratur, pusing kepala. Penyakit otot jantung disebabkan oleh tekanan darah yang tinggi(hipertensi), masalah katup jantung, kerusakan jaringan jantung dari serangan jantung sebelumnya, detak jantung yang terlalu cepat, gangguan metabolisme, kekurangan gizi, vitamin atau mineral esensial, penyalahgunaan obat kokain atau anti defrasan, penggunaan beberapa obat kemotrapi, penumpukan besi di otot jantung dan mengkonsumsi alkohol. (dr. Djoko Maryono,2008)

2.1.4.6 Penyakit Gagal Jantung

Penyakit gagal jantung adalah kondisi dimana jantung kehilangan kemampuan untuk memompa cukup darah ke jaringan tubuh. Adapun gejala-gejala yang dapat dirasakan yaitu : nafas pendek, sering batuk, pembengkakan di sekitar kaki, bengkak atau nyeri perut, mudah lelah, pusing, sesak nafas. Penyakit gagal jantung disebabkan oleh tekanan darah tinggi, anemia, kolesterol, diabetes, obesitas, merokok, obat antivirus (*zidofudin*), pola hidup tidak teratur, kurang berolahraga dan stress,meningkatnya asupan garam, endokarditis infektif, obat kemotrapi seperti doxorubicin. Penyakit ini dapat dicegah dengan mengurangi makanan berlemak, berhenti merokok, olahraga, pola hidup yang teratur dan rajin memeriksakan diri kedokter. (dr. Djoko Maryono,2008)

2.2 Aplikasi-aplikasi jantung yang sudah ada sebelumnya

Pada bab ini akan membahas tentang contoh-contoh sistem pakar penyakit jantung yang telah dibangun sebelumnya. *Contoh aplikasi sistem pakar berbasis web untuk mendiagnosa awal penyakit jantung oleh Leo willyanto santoso dkk. Aplikasi ini dibuat berbasis web dengan menggunakan php, database MySQL dan metode backward chaining. Sistem pakar ini akan mengajukan beberapa pertanyaan dan setelah semua terjawab, maka akan tampak hasil diagnosa beserta saran yang dapat membantu mengantisipasi penyakit jantung tersebut. Untuk user yang sudah terdaftar dapat melaukan tanya jawab dengan pakar melalui forum.*

Aplikasi untuk mendiagnosa penyakit jantung oleh Indrias rosiawati. Aplikasi ini dibangun dengan menggunakan mesin inferensi forward chaining dan backward chaining yang dikembangkan menggunakan bahasa pemrograman Delphi 07 dan basis data Microsoft Access 2003. Aplikasi ini dibangun untuk mengetahui jenis penyakit jantung berdasarkan gejala kemudian merekomendasikan saran dan pencegahannya. Aplikasi ini tidak akan membahas mengenai jantung secara mendalam hanya jenis penyakit jantung yang dikenal oleh masyarakat awam.

Sistem pakar diagnosa awal penyakit jantung menggunakan metode *certainty factor* berbasis android oleh Ardhito patera. Sistem ini memberikan data yang berupa kemungkinan dari jenis penyakit yang dialami, presentase keyakinan yang telah diberi oleh pakar dalam menjawab dari solusi gejala yang diisi. Dengan menggunakan perangkat mobile berbasis android dapat memberi sebuah informasi

dan mendiagnosa awal penyakit jantung pada remaja dan orangtua berdasarkan hasil hitungan dari sebuah hipotesa.

Prototipe alat deteksi dini dan mandiri penyakit jantung menggunakan sistem pakar *VICRS*, *Arduino* dan handphone android oleh Fadilla zennifa dkk. Aplikasi ini merupakan kombinasi antara sistem pakar *Variable Centered Intelligence Rule System (VCIRS)*, sensor pulsa dan *Arduino* serta handphone berbasis android yang bertujuan untuk memudahkan pengguna untuk mendeteksi penyakit jantung secara dini dan mandiri. Keluaran dari sistem ini adalah pernyataan kesimpulan bahwa pengguna didiagnosis terkena penyakit jantung atau tidak terkena kemudian dilanjutkan dengan analisa nilai *Variable Usage Rate (VUR)*, *Rule Usage Rate (RUR)* dan *Node Usage Rate (NUR)* yang menunjukkan nilai rule yang akan semakin meningkat bila sering digunakan. Hasil diagnosis yang dilakukan oleh sistem pakar sama dengan hasil diagnosis yang dilakukan oleh dokter atau pakar. Sistem ini ffokus menggunakan satu metode saja yaitu *VICRS*, perlu untuk menggabungkan dengan metode lain karena metode ini hanya menunjukkan hasil diagnosa berdasarkan rule yang telah dibuat.

2.3 Pelacakan Ke Depan (*Forward Chaining*)

Metode *forward chaining* melakukan pemrosesan berawal dari sekumpulan data untuk kemudian dilakukan inferensi sesuai dengan aturan yang diterapkan hingga diketemukan kesimpulan yang optimal. Mesin inferensi akan terus melakukan looping pada prosesnya untuk mencapai hasil keputusan yang sesuai. (Kusumadewi, 2003)

Berikut ini contoh kasus metode forward chaining :

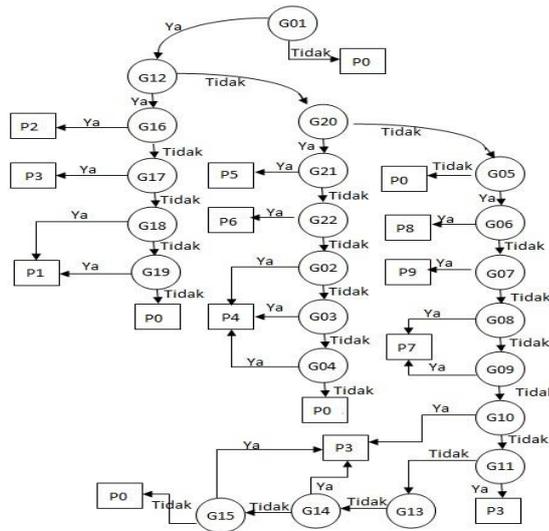
Tabel 2.1 Tabel Keputusan mengidentifikasi gejala dan penyakit mata:

Kode	P1	P2	P3	P4	P5	P6	P7	P8	P9
G1	√	√	√	√	√	√	√	√	√
G2				√					
G3				√					
G4		√		√			√	√	
G5							√		
G6			√			√			√
G7			√			√			√
G8							√		
G9							√		
G10							√		
G11							√		
G12	√	√	√						
G13							√		
G14							√		
G15							√		
G16		√		√			√	√	
G17			√			√			√
G18	√								
G19	√								
G20				√	√	√			
G21			√		√	√			√
G22			√			√			√

Keterangan:

G1...dst = Gejala 1 dan seterusnya

P 1...dst = hasil penyakit 1 dan seterusnya



Gambar 2.1 Pohon Pelacakan (Kusumadewi,2003)

2.4 Logika Fuzzy

Logika *fuzzy* didasarkan pada logika Boolean yang umum digunakan dalam komputasi. Secara ringkas, teorema *fuzzy* memungkinkan komputer “berpikir” tidak hanya dalam skala hitam-putih (0 dan 1, mati atau hidup) tetapi juga dalam skala abu-abu. Dalam Logika *Fuzzy* suatu *preposisi* dapat direpresentasikan dalam derajat kebenaran (*truthfulness*) atau kesalahan (*falsehood*) tertentu.

Pada sistem diagnosis fuzzy peranan manusia/operator lebih dominan. Pengiriman data dilaksanakan oleh operator ke dalam sistem. Ketika sistem memerlukan data tambahan. Selain itu operator dapat meminta atau menanyakan informasi dari sistem diagnosis berupa hasil konklusi atau prosedur detail hasil diagnosis oleh sistem. Dari sifat sistem ini, sistem diagnosis *fuzzy* dapat digolongkan pada sistem pakar *fuzzy*. Sistem pakar *fuzzy* adalah sistem pakar yang menggunakan notasi *fuzzy* pada aturan-aturan dan proses inference (logika keputusan).

Banyak sistem yang terlalu kompleks untuk dimodelkan secara akurat, meskipun dengan persamaan matematis yang kompleks. Dalam kasus seperti itu, ungkapan bahasa yang digunakan dalam logika kabur dapat membantu mendefinisikan karakteristik operasional sistem dengan lebih baik. Ungkapan bahasa untuk karakteristik sistem biasanya dinyatakan dalam bentuk implikasi logika. Misalnya aturan IF-THEN.

Penerapan logika fuzzy dapat meningkatkan kinerja sistem kendali dengan menekan munculnya fungsi-fungsi liar pada keluaran yang disebabkan oleh fluktuasi pada variabel masukan. Pendekatan logika fuzzy secara garis besar diimplementasikan dalam tiga tahapan yaitu :

1. Tahap pengaburan (*fuzzification*) yakni pemetaan dari masukan tegas ke himpunan kabur.
2. Tahap inferensi, yakni pembangkitan aturan kabur.
3. Tahap penegasan (*defuzzification*), yakni tranformasi keluaran dari nilai kabur ke nilai tegas.

2.4.1 Jenis Logika *Fuzzy*

Ada tiga metode dalam sistem inferensi *fuzzy* yang dapat digunakan untuk menentukan jumlah produksi, yaitu: metode *Tsukamoto*, metode *Mamdani*, dan metode *Sugeno* (Setiadji, 2009). Penjelasan mengenai ketiga metode tersebut adalah sebagai berikut:

1. Metode *Tsukamoto*

Pada metode *Tsukamoto*, setiap aturan direpresentasikan menggunakan himpunan-himpunan *fuzzy*, dengan fungsi keanggotaan yang monoton. Untuk

menentukan nilai output *crisp*/hasil yang tegas (Z) dicari dengan cara mengubah input (berupa himpunan *fuzzy* yang diperoleh dari komposisi aturan-aturan *fuzzy*) menjadi suatu bilangan pada domain himpunan *fuzzy* tersebut. Cara ini disebut dengan metode *defuzzifikasi* (penegasan). Metode *defuzzifikasi* yang digunakan dalam metode *Tsukamoto* adalah metode *defuzzifikasi* rata-rata terpusat (*Center Average Defuzzifier*).

2. Metode *Mamdani*(*Min-Max*)

Untuk metode ini, pada setiap aturan yang berbentuk implikasi (“sebab-akibat”) *anteseden* yang berbentuk konjungsi (*AND*) mempunyai nilai keanggotaan berbentuk minimum (*min*), sedangkan konsekuen gabungannya berbentuk maksimum (*max*), karena himpunan aturan-aturannya bersifat independen (tidak saling bergantung).

3. Metode *Takagi-Sugeno*

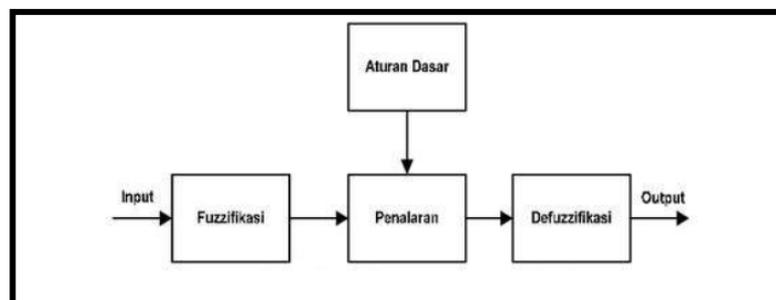
Metode *Takagi-Sugeno* adalah metode dengan mengasumsikan suatu sistem dengan m input, yaitu x_1, x_2, \dots, x_m dan satu output, yaitu Y . Metode *fuzzy* dari sistem ini terdiri atas basis aturan dengan n aturan penarikan kesimpulan *fuzzy*. Metode yang akan digunakan dalam pengambilan keputusan untuk menentukan jumlah produksi adalah metode *Tsukamoto*.

Metode ini dipilih karena setiap konsekuen pada aturan yang berbentuk *IF-THEN* direpresentasikan dengan himpunan *fuzzy* dengan fungsi keanggotaan yang monoton. Sebagai hasilnya, output dari setiap aturan diberikan secara tegas berdasarkan α , kemudian diperoleh hasil akhir dengan menggunakan rata-rata terpusat.

Metode tersebut akan digunakan untuk menentukan jumlah produksi berdasarkan data persediaan barang dan jumlah permintaan. Data persediaan barang dan jumlah permintaan adalah variabel-variabel yang akan direpresentasikan dengan fungsi keanggotaan *fuzzy*.

Untuk mempermudah pekerjaan, dalam hal ini untuk menghemat waktu dan memperkecil kesalahan dalam perhitungan, selanjutnya metode *FIS Tsukamoto* untuk menentukan jumlah produksi diterapkan dalam Sistem Pendukung Keputusan (SPK). Sehingga pembuat keputusan cukup menginputkan data-data yang diperlukan oleh SPK, yang selanjutnya disebut variabel input, yaitu: hari di mulainya produksi, masa produksi, persediaan barang maksimum satu periode tertentu, persediaan barang minimum satu periode tertentu, permintaan maksimum satu periode tertentu, permintaan minimum satu periode tertentu, produksi maksimum satu periode tertentu, produksi minimum satu periode tertentu, permintaan saat ini, dan persediaan saat ini. Kemudian SPK akan mengolah data-data tersebut dengan metode *tsukamoto* dan akan menampilkan keluaran (*output*) berupa jumlah barang yang akan diproduksi.

2.4.2 Struktur Dasar Logika *Fuzzy*



Gambar 2.2 Blok Diagram Logika *Fuzzy* (KusumaDewi, 2010)

Berdasarkan gambar 2.2, dalam system logika *fuzzy* terdapat beberapa tahapan operasional yang meliputi:

a. Fuzzifikasi

Fuzzifikasi adalah suatu proses perubahan nilai tegas yang ada ke dalam fungsi keanggotaan.

b. Penalaran (*Inference Machine*)

Mesin penalaran adalah proses implikasi dalam menalar nilai masukan guna penentuan nilai keluaran sebagai bentuk pengambilan keputusan. Salah satu model penalaran yang banyak dipakai adalah penalaran maxmin. Dalam penalaran ini, proses pertama yang dilakukan adalah melakukan operasi min sinyal keluaran lapisan fuzzifikasi, yang diteruskan dengan operasi max untuk mencari nilai keluaran yang selanjutnya akan didefuzzifikasikan sebagai bentuk keluaran.

c. Aturan Dasar (*Rule Based*)

Aturan dasar (*rule based*) pada control logika *fuzzy* merupakan suatu bentuk aturan relasi “Jika-Maka” atau “if-then” seperti berikut ini:

if x is A then y is B dimana A dan B adalah *linguistic values* yang didefinisikan dalam rentang variabel X dan Y . Pernyataan “ x is A ” disebut *antecedent* atau premis. Pernyataan “ y is B ” disebut *consequent* atau kesimpulan.

d. Defuzzifikasi

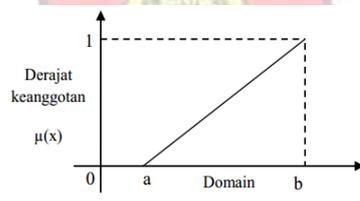
Input dari proses defuzzifikasi adalah suatu himpunan *fuzzy* yang diperoleh dari komposisi aturan-aturan *fuzzy*, sedangkan *output* yang dihasilkan merupakan suatu bilangan pada domain himpunan *fuzzy* tersebut. Sehingga jika diberikan suatu himpunan *fuzzy* dalam *range* tertentu, maka harus dapat diambil suatu nilai *crisp* tertentu.

2.4.3 Fungsi Keanggotaan

Fungsi keanggotaan adalah suatu kurva yang menunjukkan pemetaan titik-titik input ke dalam nilai keanggotaan yang memiliki interval 0 sampai 1. Salah satu cara yang dapat digunakan untuk mendapatkan nilai keanggotaan adalah dengan melalui pendekatan fungsi. Ada beberapa fungsi yang dapat digunakan, yaitu :

a. Representasi linier

Pada representasi linier, pemetaan input ke derajat keanggotaannya digambarkan sebagai garis lurus. Ada 2 keadaan himpunan fuzzy yang linier. Pertama, kenaikan himpunan dimulai pada nilai domain yang memiliki derajat keanggotaan nol [0] bergerak ke kanan menuju nilai domain yang memiliki derajat keanggotaan lebih tinggi.

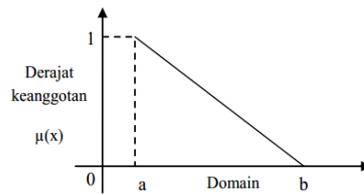


Gambar 2.3 Representasi Linier Naik (Kusumadewi dan Purnomo,2010)

Fungsi keanggotaan :

$$\mu[x] = \begin{cases} 0; & x \leq a \\ \frac{(x-a)}{(b-a)}; & a \leq x \leq b \\ 1; & x \geq b \end{cases}$$

Kedua, Garis lurus dimulai dari nilai domain dengan derajat keanggotaan tertinggi pada sisi kiri, kemudian bergerak menurun ke nilai domain yang memiliki derajat keanggotaan lebih rendah.



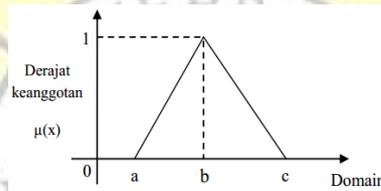
Gambar 2.4 Representasi Linier Turun (Kusumadewi dan Purnomo,2010)

Fungsi keanggotaan :

$$\mu[x] = \begin{cases} \frac{(b-x)}{(b-a)}; & a \leq x \leq b \\ 0; & x \geq b \end{cases}$$

b. Representasi kurva segitiga

Kurva segitiga pada dasarnya merupakan gabungan antara 2 garis (linier) seperti terlihat pada gambar 2.3 di bawah ini.



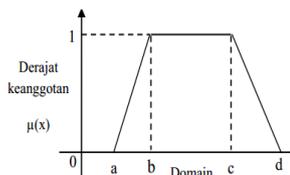
Gambar 2.5 Representasi Kurva Segitiga (Kusumadewi dan Purnomo,2010)

Fungsi keanggotaan:

$$\mu[x] = \begin{cases} 0; & x \leq a \text{ atau } x \geq c \\ (x-a)/(b-a); & a \leq x \leq b \\ (b-x)/(c-b); & b \leq x \leq c \end{cases}$$

c. Representasi kurva trapesium

Kurva trapesium pada dasarnya seperti bentuk segitiga , hanya saja pada rentang tertentu ada beberapa titik yang memiliki nilai keanggotaan 1.



Gambar 2.6 Representasi Kurva Trapesium (Kusumadewi dan Purnomo,2010)

Fungsi keanggotaan :

$$\mu[x] = \begin{cases} 0; & x \leq a \text{ atau } x \geq d \\ (x-a)/(b-a); & a \leq x \leq b \\ 1; & b \leq x \leq c \\ (d-x)/(d-c); & x \geq d \end{cases}$$

2.5 Metode Tsukamoto

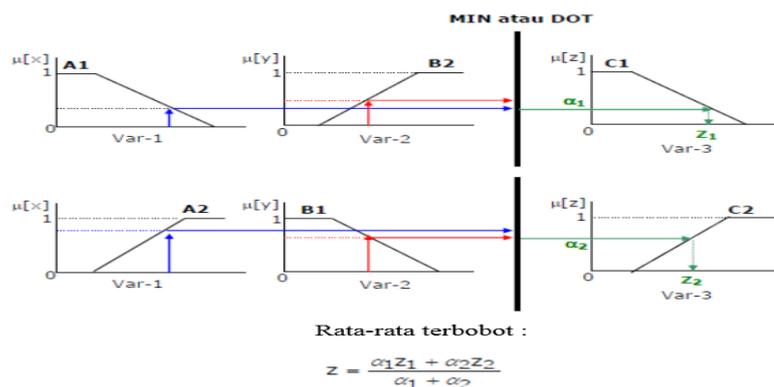
Pada metode Tsukamoto, implikasi setiap aturan berbentuk implikasi “Sebab-Akibat”/Implikasi “Input-Output” dimana antara anteseden dan konsekuen harus ada hubungannya. Setiap aturan direpresentasikan menggunakan himpunan-himpunan *fuzzy*, dengan fungsi keanggotaan yang monoton. Kemudian untuk menentukan hasil tegas digunakan rumus penegasan (*defuzifikasi*) yang disebut “Metode defuzifikasi rata-rata terpusat”. (Kusumadewi dan Purnomo, 2010). Untuk lebih memahami metode Tsukamoto, perhatikan contoh:

Misalkan ada 2 variabel input, Var-1 (x) dan Var-2(x), serta variabel output, Var-3(z), dimana Var-1 terbagi atas 2 himpunan yaitu A1 dan A2. Var-2 terbagi atas 2 himpunan B1 dan B2, Var-3 juga terbagi atas 2 himpunan yaitu C1 dan C2 (C1 dan C2 harus monoton). Ada 2 aturan yang digunakan, yaitu:

[R1] IF (A1) and (B2) THEN (C1)

[R2] IF (A2) and (B1) THEN (C2)

Pertama-tama dicari fungsi keanggotaan dari masing-masing himpunan fuzzy dari setiap aturan, yaitu himpunan A1, B2 dan C1 dari aturan fuzzy [R1], dan himpunan A2, B1 dan C2 dari aturan fuzzy [R2]. Aturan fuzzy R1 dan R2 dapat direpresentasikan dalam Gambar 2.10 untuk mendapatkan suatu nilai crisp.



Gambar 2.7 Inferensi Metode Tsukamoto (Kusumadewi dan Purnomo, 2010)

Karena pada metode Tsukamoto operasi himpunan yang digunakan adalah konjungsi (*AND*), maka nilai keanggotaan anteseden dari aturan *fuzzy*[R1] adalah irisan dari nilai keanggotaan A1 dari Var-1 dengan nilai keanggotaan B1 dari Var-2. Menurut teori operasi himpunan pada persamaan, maka nilai keanggotaan anteseden dari operasi konjungsi (*And*) dari aturan *fuzzy*[R1] adalah nilai minimum antara nilai keanggotaan A1 dari Var-1 dan nilai keanggotaan B2 dari Var-2.

Demikian pula nilai keanggotaan anteseden dari aturan *fuzzy*[R2] adalah nilai minimum antara nilai keanggotaan A2 dari Var-1 dengan nilai keanggotaan B1 dari Var-2. Selanjutnya, nilai keanggotaan *anteseden* dari aturan *fuzzy*[R1] dan [R2] masing-masing disebut dengan α_1 dan α_2 . Nilai α_1 dan α_2 kemudian disubstitusikan pada fungsi keanggotaan himpunan C1 dan C2 sesuai aturan *fuzzy* [R1] dan [R2] untuk memperoleh nilai z_1 dan z_2 , yaitu nilai z (nilai perkiraan produksi) untuk aturan *fuzzy*[R1] dan [R2].

Untuk memperoleh nilai output crisp/nilai tegas Z , dicari dengan cara mengubah input (berupa himpunan *fuzzy* yang diperoleh dari komposisi aturan-aturan *fuzzy*) menjadi suatu bilangan pada domain himpunan *fuzzy* tersebut. Cara ini disebut dengan metode *defuzifikasi* (penegasan).

2.6 Pengenalan Android

Android merupakan sebuah system operasi telepon seluler dan computer tablet layar sentuh(touchscreen) yang berisi Linux.

Namun android berubah menjadi platform yang begitu cepat dalam melakukan inovasi. Hal ini tidak lepas dari pengembang utama dibelakangnya yaitu Google. Platform android terdiri dari system operasi berbasis Linux, sebuah GUI (Graphic User Interface), sebuah web browser dan aplikasi end-user yang dapat di download dan para pengembang bisa leluasa berkarya menciptakan aplikasi yang terbaik untuk digunakan oleh berbagai macam perangkat. (Akhmad Dharma Kasman, 2013)

2.6.1 IDE Eclipse

Eclipse merupakan program untuk mengetikkan baris perintah untuk membuat aplikasi android, script program yang diketikkan di eclipse merupakan perpaduan antara bahasa pemrograman java dan xml. Eclipse adalah sebuah IDE (Integrated Development Environment) untuk mengembangkan perangkat lunak dan dapat dijalankan di semua platform (platform-independent).

Berikut ini adalah sifat dari Eclipse :

1. Multi-platform, Target sistem operasi Eclipse adalah Microsoft Windows, Linux, Solaris, AIX, HP-UX dan Mac OS X.
2. Multi- Language, Eclipse dikembangkan dengan bahasa pemrograman Java, akan tetapi Eclipse mendukung pengembangan aplikasi berbasis bahasa pemrograman lainnya, seperti C/C++, Cobol, Phyton, Perl, PHP, dan lain sebagainya.

3. Multi-role, Selain sebagai IDE untuk pengembangan aplikasi, Eclipse pun bisa digunakan untuk aktivitas dalam siklus pengembangan perangkat lunak, seperti dokumentasi, test perangkat lunak, pengembangan web, dan lain sebagainya. (Akhmad Dharma Kasman, 2013)

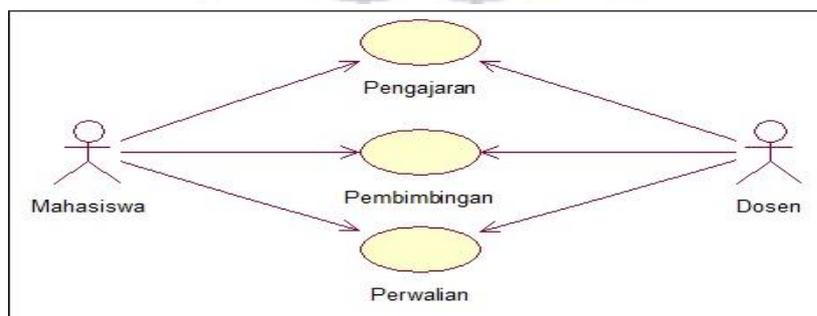
2.7 UML

UML (Unified Modeling Language) adalah ‘bahasa’ pemodelan untuk sistem atau perangkat lunak yang berparadigma ‘berorientasi objek’. Pemodelan digunakan untuk penyederhanaan permasalahan yang kompleks sehingga lebih mudah dipahami. (Adi Nugroho, 2010)

2.7.1 Usecase Diagram

Usecase digunakan untuk mendefinisikan suatu bagian perilaku sistem yang bersifat koheren tanpa perlu menyingkapkan struktur internal sistem/perangkat lunak yang sedang dikembangkan. (Adi Nugroho, 2010)

Berikut contoh sederhana *usecasediagram* pada gambar 2.8

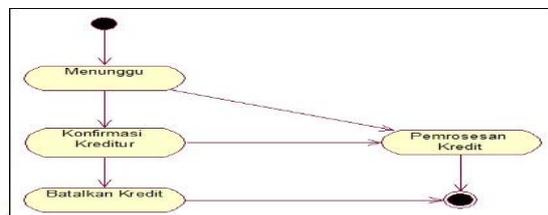


Gambar 2.8 Contoh *Usecase Diagram* (Adi Nugroho, 2010)

2.7.2 Activity Diagram

Activity diagram adalah view untuk suatu objek yang terokalisasi, suatu view yang memisahkannya dari lingkungan luar dan melakukan pengujian perilaku dalam keadaan terisolasi. Suatu diagram aktivitas memuat di dalamnya *activity state* dimana suatu *activity state* mempresentasikan eksekusi pernyataan dalam suatu prosedur atau kinerja suatu aktivitas dalam suatu aliran kerja.

Berikut contoh sederhana *activity diagram* pada gambar 2.9

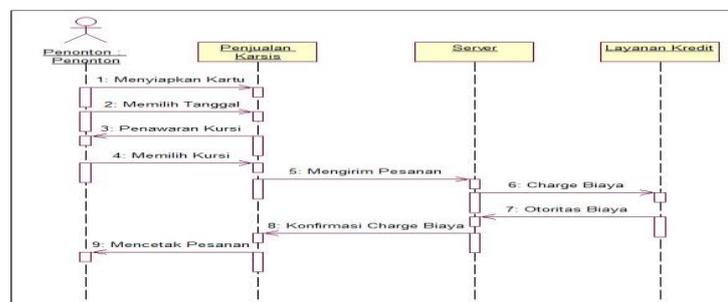


Gambar 2.9 Contoh *Activity Diagram*(Adi Nugroho, 2010)

2.7.3 Sequence Diagram

Sequence diagram menjelaskan secara detail urutan proses yang dilakukan dalam sistem untuk mencapai tujuan dari *usecase* : interaksi yang terjadi antar class, operasi apa saja yang terlibat, urutan antar operasi dan informasi yang diperlukan oleh masing-masing operasi. (Adi Nugroho, 2010)

Berikut contoh sederhana *sequence diagram* pada gambar 2.10



Gambar 2.10 Contoh *Sequence Diagram*(Adi Nugroho, 2010)