

BAB II

LANDASAN TEORI

2.1 *Decision Support System (DSS)*

Decision Support System (DSS) dapat dikatakan sebagai sistem pendukung keputusan yang merupakan bagian dari sistem informasi berbasis komputer yang digunakan untuk mendukung bisnis atau kegiatan pengambilan keputusan dalam suatu organisasi atau perusahaan. *Decision Support system (DSS)* merupakan bagian dari sistem informasi berbasis komputer termasuk sistem berbasis pengetahuan (*knowledge management*) yang digunakan untuk mendukung pengambilan keputusan dalam suatu organisasi atau perusahaan. Implementasi teori-teori pengambilan keputusan yang telah diperkenalkan oleh ilmu-ilmu seperti *operation research* dan *management science*, hanya bedanya adalah bahwa jika dahulu untuk mencari penyelesaian masalah yang dihadapi harus dilakukan perhitungan iterasi secara manual (biasanya untuk mencari nilai minimum, maksimum, atau optimum), saat ini komputer *PC* telah menawarkan kemampuannya untuk menyelesaikan persoalan yang sama dalam waktu relatif singkat Penyusunan model keputusan adalah suatu cara untuk mengembangkan hubungan logis yang mendasari persoalan keputusan ke dalam bentuk matematis yang mencerminkan hubungan yang terjadi antara faktor yang terlibat. Hal yang paling sulit dilakukan setelah keputusan didapat adalah segi penerapannya karena perlu menyakinkan semua orang yang terlibat, bahwa keputusan adalah yang terbaik (Suryadi, 1998).

2.1.1 Karakteristik Sistem Pendukung Keputusan (SPK)

Decision Support system (DSS) memiliki 6 karakteristik antara lain sebagai berikut :

1. Mendukung proses pengambilan keputusan yang menitik beratkan pada manajemen dengan persepsi.
2. Adanya *interface* manusia atau mesin dimana manusia sebagai user tetap memegang kontrol proses pengambilan keputusan.
3. Mendukung pengambilan keputusan untuk membahas masalah terstruktur, semi terstruktur dan tidak terstruktur.
4. Memiliki kapasitas dialog untuk memperoleh informasi sesuai dengan kebutuhan.
5. Memiliki subsistem-subsistem yang terintegrasi sedemikian rupa sehingga dapat berfungsi sebagai kesatuan sistem.
6. Membutuhkan struktur data komprehensif yang dapat melayani kebutuhan informasi seluruh tahap manajemen

2.1.2 Komponen *Decision Support System* (DSS)

Berikut untuk komponen yang digunakan pada proses penentuan keputusan antara lain :

a. *Database Management*

Merupakan subsistem data yang terorganisasi dalam suatu basis data. Data yang merupakan suatu sistem pendukung keputusan dapat berasal dari luar maupun dalam lingkungan. Untuk keperluan SPK, diperlukan data yang relevan dengan permasalahan yang hendak dipecahkan melalui simulasi.

b. *Model Base*

Merupakan suatu model yang merepresentasikan permasalahan kedalam format kuantitatif (model matematika sebagai contohnya) sebagai dasar simulasi atau pengambilan keputusan, termasuk didalamnya tujuan dari permasalahan (*objektif*), komponen-komponen terkait, batasan-batasan yang ada (*constraints*), dan hal-hal terkait lainnya. *Model Base* memungkinkan pengambil keputusan menganalisa secara utuh dengan mengembangkan dan membandingkan solusi alternatif.

c. *User Interface / Pengelolaan Dialog*

Terkadang disebut sebagai subsistem dialog, merupakan penggabungan antara dua komponen sebelumnya yaitu *Database Management* dan *Model Base* yang disatukan dalam komponen ketiga (*user interface*), setelah sebelumnya dipresentasikan dalam bentuk model yang dimengerti komputer. *User Interface* menampilkan keluaran sistem bagi pemakai dan menerima masukan dari pemakai kedalam Sistem Pendukung Keputusan.

Ada 4 tahapan yang perlu diperhatikan dalam *Decision Support System (DSS)*, diantaranya:

1. *Intelligence* : Mencari kondisi yang akan digunakan untuk memanggil/meminta sebuah keputusan
2. *Design* : Mengembangkan dan menganalisis alternatif tindakan yang memungkinkan dari solusi.
3. *Choice* : Memilih tindakan diantara 2 tahap sebelumnya.
4. *Implementation* : Memakai tindakan yang dipilih dalam tindakan situasi pengambilan keputusan.

2.1.3 Manfaat *Decision Support System (DSS)*

Decision Support System (DSS), dapat memberikan berbagai manfaat dan keuntungan. Manfaat yang dapat diambil dari SPK adalah :

1. Memperluas kemampuan pengambil keputusan dalam memproses data / informasi bagi pemakainya.
2. Membantu pengambil keputusan untuk memecahkan masalah terutama berbagai masalah yang sangat kompleks dan tidak terstruktur.
3. Dapat menghasilkan solusi dengan lebih cepat serta hasilnya dapat diandalkan.
4. Walaupun suatu *Decision Support System (DSS)* mungkin saja tidak mampu memecahkan masalah yang dihadapi oleh pengambil keputusan, namun dia dapat menjadi stimulan pengambil keputusan

memahami persoalannya, karena mampu menyajikan berbagai alternatif pemecahan.

2.2 Pengertian SISWA

Siswa adalah salah satu komponen manusiawi yang menempati posisi sentral dalam proses belajar mengajar dimana di dalam proses belajar mengajar, siswa sebagai pihak yang ingin meraih cita-cita, memiliki tujuan dan kemudian ingin mencapainya secara optimal. Siswa akan menjadi faktor penentu, sehingga dapat mempengaruhi segala sesuatu yang diperlukan untuk mencapai tujuan belajarnya. Pengertian siswa dalam Kamus Bahasa Indonesia adalah orang/anak yang sedang berguru (belajar, bersekolah). Menurut Prof. Dr. Shafique Ali Khan (2005) pengertian siswa adalah orang yang datang ke suatu lembaga untuk memperoleh atau mempelajari beberapa tipe pendidikan. Sedangkan menurut Daradjat (1995) siswa adalah pribadi yang “unik” yang mempunyai potensi dan mengalami proses berkembang. Dalam proses berkembang itu siswa membutuhkan bantuan yang sifat dan contohnya tidak ditentukan oleh guru tetapi oleh anak itu sendiri, dalam suatu kehidupan bersama dengan individu-individu yang lain.

Murid atau anak didik adalah salah satu komponen manusiawi yang menempati posisi sentral dalam proses belajar-mengajar. Di dalam proses belajar-mengajar, murid sebagai pihak yang ingin meraih cita-cita, memiliki tujuan dan kemudian ingin mencapainya secara optimal. Murid akan menjadi faktor penentu, sehingga dapat mempengaruhi segala sesuatu yang diperlukan untuk mencapai tujuan belajarnya. Murid atau anak adalah pribadi yang “unik” yang mempunyai potensi dan mengalami proses berkembang. Dalam proses berkembang itu anak atau murid membutuhkan bantuan yang sifat dan coraknya tidak ditentukan oleh guru tetapi oleh anak itu sendiri, dalam suatu kehidupan bersama dengan individu-individu yang lain. Tugas Siswa / Murid / Peserta Didik

Selain guru, murid pun mempunyai tugas untuk menjaga hubungan baik dengan guru maupun dengan sesama temannya dan untuk senantiasa

meningkatkan keefektifan belajar bagi kepentingan dirinya sendiri. Adapun tugas tersebut ditinjau dari berbagai aspek yaitu aspek yang berhubungan dengan belajar, aspek yang berhubungan dengan bimbingan, dan aspek yang berhubungan dengan administrasi. Aspek yang berhubungan dengan belajar kesalahan-kesalahan dalam belajar sering dilakukan murid, bukan saja karena ketidaktahuannya, tetapi juga disebabkan oleh kebiasaan-kebiasaannya yang salah. Adalah menjadi tugas murid untuk belajar baik yang menghindari atau mengubah cara-cara yang salah itu agar tercapai hasil belajar yang maksimal. Hal-hal yang harus diperhatikan murid agar belajar menjadi efektif dan produktif, di antaranya:

1. Murid harus menyadari sepenuhnya akan arah dan tujuan belajarnya, sehingga ia senantiasa siap siaga untuk menerima dan mencernakan bahan. Jadi bukan belajar asal belajar saja. Murid harus memiliki motif yang murni (intrinsik atau niat).
2. Harus belajar dengan “kepala penuh”, artinya murid memiliki pengetahuan dan pengalaman-pengalaman belajar sebelumnya (apersepsi), sehingga memudahkan dirinya untuk menerima sesuatu yang baru. Murid harus menyadari bahwa belajar bukan semata-mata menghafal. Di dalamnya juga terdapat penggunaan daya-daya mental lainnya yang harus dikembangkan sehingga memungkinkan dirinya memperoleh pengalaman-pengalaman baru dan mampu memecahkan berbagai masalah.
3. Harus senantiasa memusatkan perhatian (konsentrasi pikiran) terhadap apa yang sedang dipelajari dan berusaha menjauhkan hal-hal yang mengganggu konsentrasi sehingga terbina suasana ketertiban dan keamanan belajar bersama dan/atau sendiri.
4. Harus memiliki rencana belajar yang jelas, sehingga terhindar dari perbuatan belajar yang “insidental”. Jadi belajar harus merupakan suatu kebutuhan dan kebiasaan yang teratur, bukan “seenaknya” saja.

5. Murid harus memandang bahwa semua ilmu (bidang studi) itu sama penting bagi dirinya, sehingga semua bidang studi dipelajarinya dengan sungguh-sungguh.
6. Jangan melalaikan waktu belajar dengan membuang-buang waktu atau bersantai-santai. Gunakan waktu seefisien mungkin dan hanya bersantai sekadar melepaskan lelah atau mengendorkan uraf saraf yang telah tegang dengan berekreasi.
7. Harus dapat bekerja sama dengan kelompok/kelas untuk mendapatkan sesuatu atau memperoleh pengalaman baru dan harus teguh bekerja sendiri dalam membuktikan keberhasilan belajar, sehingga ia tahu benar akan batas-batas kemampuannya.
8. Selama mengikuti pelajaran atau diskusi dalam kelompok/kelas, harus menunjukkan partisipasi aktif dengan jalan bertanya atau mengeluarkan pendapat, bila diperlukan

2.3 Pengertian *MySQL*

MySQL adalah salah satu aplikasi RDBMS (*Relational Database Management System*). Pengertian sederhana RDBMS adalah: aplikasi *database* yang menggunakan prinsip relasional. Dalam tutorial berikutnya DBMS dibedakan menjadi dua kategori, yaitu DBMS berbasis file, dan DBMS berbasis *client-server*. DBMS berbasis file (contohnya *Microsoft Access* dan *FileMaker*) dirancang untuk penggunaan berbasis desktop dan bukan dimaksudkan untuk digunakan dalam aplikasi-aplikasi kritis atau higher end.

MySQL ini tergolong suatu software yang open source dan berlisensi GPL atau *General Public License*. Lisensi GPL ini hanya ditujukan pada perangkat lunak tertentu untuk keperluan proyek GNU, inilah yang menjadi faktor banyaknya pengguna *MySQL* di seluruh dunia. Selain mudah digunakan, anda dapat mengelola data dengan lebih efektif karena menggunakan *script* atau Bahasa tertentu dan secara otomatis akan menjadi perintah ke *system*

2.3.1 PHP (*Personal Home Page*)

PHP adalah *bahasa pemrograman script server-side yang didesain untuk pengembangan web*. Selain itu, *PHP juga bisa digunakan sebagai bahasa pemrograman umum* ([wikipedia](#)). PHP dikembangkan pada tahun 1995 oleh Rasmus Lerdorf, PHP merupakan singkatan dari *Personal Home Page*. Sesuai dengan namanya, PHP digunakan untuk membuat website pribadi. Dalam beberapa tahun perkembangannya, PHP menjelma menjadi bahasa pemrograman web yang powerful dan tidak hanya digunakan untuk membuat halaman web sederhana, tetapi juga website populer yang digunakan oleh jutaan orang seperti wikipedia, wordpress, joomla, dll Saat ini PHP adalah singkatan dari PHP: Hypertext Preprocessor, sebuah kepanjangan *rekursif*, yakni permainan kata dimana kepanjangannya terdiri dari singkatan itu sendiri: PHP: Hypertext Preprocessor. PHP dapat digunakan dengan gratis (*free*) dan bersifat *Open Source*. PHP dirilis dalam lisensi *PHP License*, sedikit berbeda dengan lisensi *GNU General Public License (GPL)* yang biasa digunakan untuk proyek *Open Source*.

2.3.2 Fungsi PHP Dalam Pemrograman Web

Sebagai perbandingan, website dinamis yang bisa dibuat menggunakan PHP adalah situs web yang bisa menyesuaikan tampilan konten tergantung situasi. Website dinamis juga bisa menyimpan data ke dalam *database*, membuat halaman yang berubah-ubah sesuai *input* dari *user*, memproses form. Untuk pembuatan *web*, kode *PHP* biasanya di sisipkan kedalam dokumen *HTML*. Karena fitur inilah *PHP* disebut juga sebagai Scripting Language atau bahasa pemrograman *script*. Halaman *HTML* tersebut dapat dibuat dengan mudah dengan cara *copy-paste tag* sebanyak 10 kali dan mengubah sedikit angka-angka no urut di belakangnya. Namun jika yang kita inginkan adalah menambahkan list tersebut menjadi 100 atau 1000 *list*, cara *copy-paste* tersebut menjadi tidak efektif

2.4 Pengenalan *Xampp*

Fungsinya dari *xampp* sendiri sebagai server yang berdiri sendiri (localhost) dan terdiri atas program Apache *HTTP Server*, *MySQL*, *database*, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman *PHP* dan *Perl*. Program ini tersedia dalam GNU *General Public License* dan bebas, merupakan *web server* yang mudah digunakan yang dapat melayani tampilan halaman *web* yang dinamis. Aplikasi yang mampu membundel apache menjadi satu mysql dan *PHP* menjadi satu kesatuan dalam *file installer* yaitu *Phptriad*. Dimana pada proses Instalasi yang sangat mudah membuat kerja web developer tidak kesulitan untuk menginstall *apache* dan *mysql* dari awal. Sayang sekali *project phptriad* ini yang dulunya adalah *project open source discontinued* dan berganti baju menjadi *project* komersial. Pada dasarnya *Phptriad* hanyalah memaketkan aplikasi-aplikasi *web developer* yang selama ini sering digunakan dalam lingkungan sistem operasi *Linux*. *LAMP* yang sudah bertahun-tahun menjadi andalan bagi web developer dalam sistem operasi linux ternyata juga bisa diimplementasikan dalam sistem operasi *windows*. Ini artinya pasar bagi *Apache*, *Mysql* dan *PHP* sudah mulai merambah ke sistem operasi selain *linux*.

Kesempatan itu tentunya tidak disia-siakan oleh pengembang lainnya. Bundel 3 aplikasi (*apache*, *mysql* dan *PHP*) mulai bertebaran setelah *Phptriad* muncul. Beberapa yang sempat tercatat dan di ketahui antara lain *FoxServ*, *WinLamp* dan *Xampp*. Dari ketiganya *Xampp* lebih memberikan kemudahan pada proses instalasi dan konfigurasi seperti yang dimiliki oleh *Phptriad*. Berikut ini adalah bagian-bagian *XAMPP* yang biasa digunakan pada umumnya:

- a. ***Htdoc*** adalah folder tempat meletakkan berkas-berkas yang akan dijalankan, seperti berkas *PHP*, *HTML* dan skrip lain.
- b. ***PhpMyAdmin*** merupakan bagian untuk mengelola basis data *MySQL* yang ada dikomputer. Untuk membukanya, buka browser lalu

ketikkan alamat <http://localhost/phpMyAdmin>, maka akan muncul halaman *phpMyAdmin*.

- c. **Control Panel** yang berfungsi untuk mengelola layanan (*service*) *XAMPP*. Seperti menghentikan (*stop*) layanan, ataupun memulai (*start*).

2.5 Logika *Fuzzy*

(Kusumadewi,2004). Logika *fuzzy* adalah suatu cara yang tepat untuk memetakan ruang *input* kedalam suatu ruang *output*. (Kusumadewi,2004). Logika *fuzzy* menggunakan ungkapan bahasa untuk menggambarkan nilai variabel. Logika *fuzzy* bekerja dengan menggunakan derajat keanggotaan dari sebuah nilai yang kemudian digunakan untuk menentukan hasil yang ingin dicapai berdasarkan atas spesifikasi yang telah ditentukan. Ada beberapa hal yang perlu diketahui dalam memahami sistem *fuzzy* yaitu: Variabel *Fuzzy*, Himpunan *Fuzzy*, Semesta Pembicaraan, dan *Domain* himpunan *fuzzy*.

2.5.1 Himpunan *Fuzzy*

Himpunan tegas (*crisp*) *A* didefinisikan oleh *item-item* yang ada pada himpunan itu. Jika $a \in A$, maka nilai yang berhubungan dengan *A* adalah 1. Namun jika *a* bukan anggota *A*, maka nilai yang berhubungan dengan *a* adalah 0. notasi $A = \{x|P(x)\}$ menunjukkan bahwa *A* berisi *item* *x* dengan *P(x)* benar. Jika *X* merupakan fungsi karakteristik *A* dan properti *P*, maka dapat dikatakan bahwa *P(x)* benar, jika dan hanya jika $X(x)=1$.

Himpunan *fuzzy* didasarkan pada gagasan untuk memperluas jangkauan fungsi karakteristik sedemikian hingga fungsi tersebut akan mencakup bilangan real pada interval $[0,1]$. Nilai keanggotaannya menunjukkan bahwa suatu *item* dalam semesta pembicaraan tidak hanya bernilai 0 atau 1, namun juga nilai yang terletak diantaranya. Dengan kata lain, nilai kebenaran suatu *item* tidak hanya benar (1)

atau salah (0) melainkan masih ada nilai-nilai yang terletak diantara benar dan salah. Himpunan *fuzzy* memiliki 2 atribut, (Kusumadewi, 2004) yaitu :

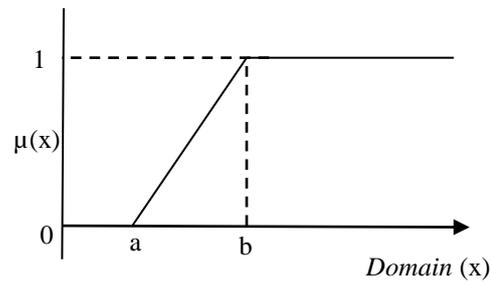
1. Linguistik yaitu penamaan suatu grup yang mewakili suatu keadaan atau kondisi tertentu dengan menggunakan bahasa alami. Contoh : murah, sedang, mahal.
2. Numeris yaitu suatu nilai (angka) yang menunjukkan ukuran dari suatu variabel. Contoh : 100, 500, 1000, dan seterusnya.

2.5.2 Fungsi Keanggotaan

Fungsi keanggotaan (*membership function*) adalah suatu kurva yang menunjukkan pemetaan titik-titik *input* data ke dalam nilai keanggotaannya (sering disebut dengan derajat keanggotaan) yang memiliki interval antara 0 sampai 1. Ada dua cara mendefinisikan keanggotaan himpunan *fuzzy*, yaitu secara numeris dan fungsional. Definisi numeris menyatakan fungsi derajat keanggotaan sebagai vector jumlah yang tergantung pada tingkat diskretisasi. Misalnya, jumlah elemen diskret dalam semesta pembicaraan. Definisi Fungsional menyatakan derajat Keanggotaan. batasan ekspresi analisis yang dapat dihitung. Standar atau ukuran tertentu pada fungsi keanggotaan secara umum berdasar atas semesta X bilangan real :

1. Representasi *Linear*

Ada 2 kemungkinan himpunan *fuzzy linear* yaitu: Kenaikan himpunan dimulai pada nilai domain yang memiliki derajat keanggotaan nol [0] bergerak kekanan menuju nilai *domain* yang memiliki derajat keanggotaan lebih tinggi. Fungsi *linear* naik (bahu kanan) dirumuskan seperti gambar 2.1 :



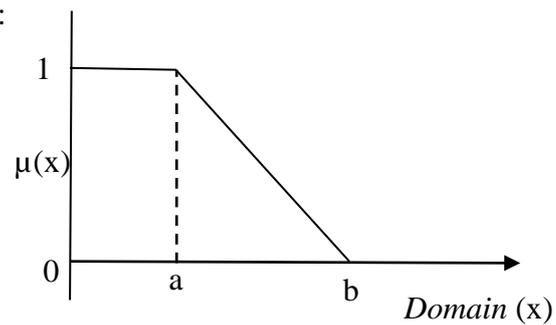
Gambar 2.5 Himpunan *Fuzzy Linear Naik*.

Persamaan 2.1

Fungsi Keanggotaan dari *linear* naik adalah

$$\left\{ \begin{array}{ll} 0; & x \leq a \\ (x-a) / (b-a) & a \leq x \leq b \\ 1; & x \geq b \end{array} \right.$$

Fungsi *linear* turun (bahu kiri) dirumuskan seperti gambar 2.2 dibawah ini:



Gambar 2.6 Himpunan *Fuzzy Linear Turun*.

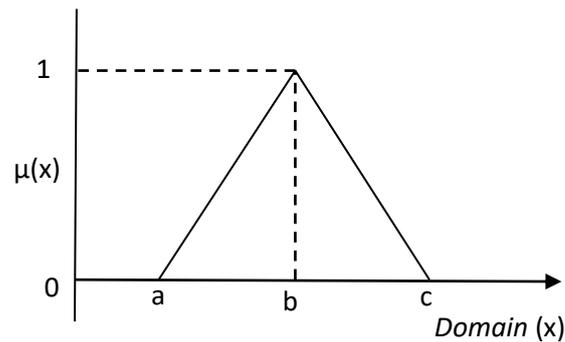
Fungsi Keanggotaan dari *linear* turun adalah

$$\left\{ \begin{array}{ll} 1; & x \leq a \\ (b-x) / (b-a) & a \leq x \leq b \\ 0; & x \geq b \end{array} \right.$$

Persamaan 2.2

2. Representasi Kurva Segitiga

Kurva segitiga pada dasarnya merupakan gabungan antara 2 garis (linier), Fungsi segitiga dirumuskan seperti gambar 2.3 dibawah ini:



Gambar 2.7 Kurva Segitiga

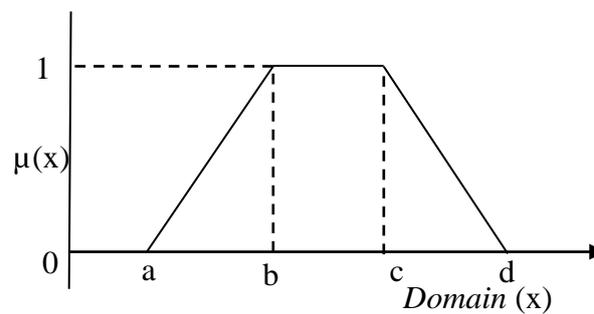
Fungsi Keanggotaan dari Segitiga adalah

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

Persamaan 2.3

3. Representasi Kurva Trapesium

Kurva segitiga pada dasarnya seperti titik yang memiliki nilai keanggotaan 1.



Gambar 2.8 Kurva Trapesium

Fungsi Keanggotaan dari Trapesium adalah

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

Persamaan 2.4

2.5.3 Fuzzy Database Model Tahani

Fuzzy tahani merupakan salah satu metode *fuzzy* yang menggunakan basis data standar. Pada basis data standar, data diklasifikasikan berdasarkan bagaimana

data tersebut dipandang oleh *user*. Oleh karena itu pada basis data standar data yang ditampilkan akan keluar seperti data yang telah disimpan. Namun pada kenyataannya, seseorang kadang membutuhkan informasi dari data-data yang bersifat ambiguous. Basis data *fuzzy* model Tahani masih tetap menggunakan relasi standar, hanya saja model ini menggunakan teori himpunan *fuzzy* untuk mendapatkan informasi pada *query*-nya. Berikut untuk tahapan Model Tahani tersusun atas :

a. Menggambarkan Fungsi Keanggotaan

Fungsi keanggotaan (*membership function*) adalah suatu kurva yang menunjukkan pemetaan titik-titik *input* data ke dalam nilai keanggotaannya (sering juga disebut dengan derajat keanggotaan) yang memiliki internal antara 0 sampai 1, salah satu cara yang dapat digunakan untuk mendapatkan nilai keanggotaan melalui pendekatan fungsi. Beberapa fungsi yang dapat digunakan yaitu Representasi kurva Linier, Representasi Kurva Segitiga. Representasi Kurva Trapesium. Representasi Kurva Bentuk Bahu, Representasi Kurva-S, Representasi Kurva Bentuk Lonceng (*Bell Curve*). Masing-masing fungsi tersebut, akan menghasilkan nilai antara “0” dan “1” dengan cara yang berbeda, sesuai dengan jenis representasi yang digunakan.

b. Fuzzifikasi

Fuzzifikasi adalah fase pertama dari perhitungan *fuzzy* yaitu perubahan nilai tegas ke nilai *fuzzy*. Prosesnya adalah sebagai berikut: Suatu besaran analog dimasukkan sebagai masukan (*crisp input*), lalu *input* tersebut dimasukkan pada batas scope dari membership function. Membership function ini biasanya dinamakan membership function *input*. Keluaran dari proses fuzzifikasi ini adalah sebuah nilai *input fuzzy* atau yang biasanya dinamakan *fuzzy input*.

c. Fuzzifikasi *Query*

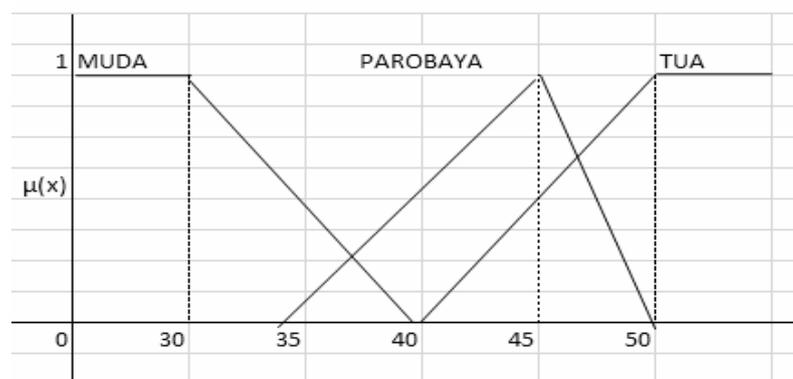
Fuzzifikasi *Query* diasumsikan sebuah *query* konvensional (*nonfuzzy*) DBMS yang akan mencoba membuat dan menerapkan sebuah sistem dasar logika *fuzzy query* .

d. Operator Dasar Zadeh untuk Operasi Himpunan *Fuzzy*.

Nilai keanggotaan sebagai dari 2 himpunan *fuzzy* dikenal dengan nama Fire Strength atau α -predikat. Sangat mungkin digunakan operator dasar dalam proses *query* berupa operator *AND* dan *OR*. α -predikat sebagai hasil operasi dengan operator *AND* diperoleh dengan mengambil nilai keanggotaan terkecil antar elemen pada himpunan-himpunan yang bersangkutan, dinotasikan : $\mu_{A \cap B} = \min(\mu_A[x], \mu_B[y])$. Sedangkan untuk hasil operasi dengan operator *OR* diperoleh dengan mengambil nilai keanggotaan terbesar antar elemen pada himpunan-himpunan yang bersangkutan, dinotasikan : $\mu_{A \cup B} = \max(\mu_A[x], \mu_B[y])$. Alternatif yang direkomendasikan adalah alternatif yang memiliki nilai Fire Strength atau tingkat kesesuaian dengan kriteria pilihan di atas angka 0 (nol) sampai dengan angka 1 (satu)

2.5.4 Pengolahan Data Karyawan Kedalam *Fuzzy Database*

Umur



Gambar 2.9 Fungsi Keanggotaan Untuk Variabel Umur

$$\text{Fungsi Keanggotaan :}$$

$$\mu_{\text{Muda}}[u] = \begin{cases} 1; & u \leq 30 \\ (40-u) / (40-30) & 30 \leq u \leq 40 \\ 0; & u \geq 40 \end{cases}$$

$$\mu_{\text{Parobaya}}[u] = \begin{cases} 0; & u \leq 30 \text{ atau } u \geq 50 \\ (u-35) / (45-35); & 35 \leq u \leq 45 \\ (50-u) / (50-45); & 45 \leq u \leq 50 \end{cases}$$

$$\mu_{\text{Tua}}[u] = \begin{cases} 0; & u \leq 40 \\ (u-40) / (40-50) & 40 \leq u \leq 50 \\ 1; & u \geq 50 \end{cases}$$

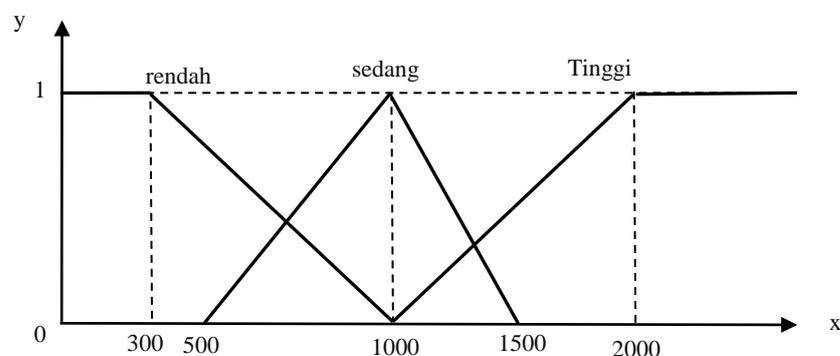
Untuk nilai dikurang dari sama dengan fungsi dari keanggotaan maka nilainya 0 dan untuk data diatas dari fungsi keanggotaan maka nilainya 1. Untuk hasil perhitungan dari 10 data berikutnya dapat dilihat pada tabel 2.4 menunjukkan tabel karyawan berdasarkan umur dengan derajat keanggotaannya pada setiap himpunan :

Tabel 2.5 Fungsi keanggotaan Berdasarkan Umur

NIP	Nama	Umur (th)	Derajat Keanggotaan $\mu[x]$		
			MUDA	PAROBAYA	TUA
01	Lia	30	1	0	0
02	Iwan	48	0	0.4	0.8
03	Sari	36	0.4	0.1	0
04	Andi	37	0.3	0.2	0
05	Budi	42	0	0.7	0.2
06	Amir	39	0.1	0.4	0
07	Rian	37	0.3	0.2	0
08	Kiki	32	0.8	0	0
09	Alda	35	0.5	0	0
10	Yoga	25	1	0	0

Gaji

Untuk Variabel Gaji bisa dikategorikan dalam himpunan: Rendah, Sedang dan tinggi terlihat pada gambar 2.6 :



Gambar 2.10 Fungsi keanggotaan untuk variabel Gaji

Fungsi Keanggotaan :

$$\mu_{\text{Rendah}}[z] = \begin{cases} 1; & z \leq 300 \\ (800-z) / (800-300) & 300 \leq z \leq 800 \\ 0; & \geq 800 \end{cases}$$

$$\mu_{\text{Sedang}}[z] = \begin{cases} 0; & z \leq 500 \text{ atau } z \geq 1500 \\ (z-500) / 500; & 500 \leq z \leq 1000 \\ (1500-z) / (500); & 1000 \leq z \leq 1500 \end{cases}$$

$$\mu_{\text{Tinggi}}[z] = \begin{cases} 0; & z \leq 1000 \\ (z-1000) / (1000) & 1000 \leq z \leq 2000 \\ 1; & \end{cases}$$

Untuk nilai kurang dari sama dengan fungsi dari keanggotaan maka nilainya 0 dan untuk data diatas dari fungsi keanggotaan maka nilainya 1. Dan untuk hasil perhitungan dari 10 data berikutnya dapat dilihat pada tabel 2.5 menunjukkan tabel karyawan berdasarkan gaji dengan derajat keanggotaannya pada setiap himpunan, berikut tabelnya :

Tabel 2.6 Karyawan Berdasarkan Gaji

NIP	Nama	Gaji/(bl)	Derajat Keanggotaan $\mu[z]$		
			Rendah	Sedang	Tinggi
01	Lia	750.000	0.1	0.5	0
02	Iwan	1.255.000	0	0.49	0.255
03	Sari	1.500.000	0	0	0.500
04	Andi	1.040.000	0	0.92	0.040
05	Budi	950.000	0	0.9	0.2
06	Amir	1.600.000	0	0	0.600
07	Rian	1.250.000	0	0.50	0.250
08	Kiki	550.000	0.5	0	0
09	Alda	735.000	0.13	0	0
10	Yoga	860.000	0	0	0

Berikut adalah Penggunaan contoh dari beberapa *query* yang bisa diberikan dari fungsi keanggotaan dengan *fuzzy database* model tahani berikut *query*-annya :

Query1:

Siapa saja-kah karyawan yang masih muda tapi memiliki gaji tinggi?

SELECT NAMA FROM KARYAWAN

WHERE (Umur = "MUDA") and (Gaji = "TINGGI")

Tabel 2.7 karyawan yang masih muda tapi memiliki gaji yang tinggi

NIP	Nama	Umur (th)	Gaji/(bl)	Derajat Keanggotaan $\mu[z]$		
				Muda	Tinggi	Muda & tinggi
01	Lia	30	750.000	1	0	0
02	Iwan	48	1.255.000	0	0.255	0
03	Sari	36	1.500.000	0.4	0.5	0.4
04	Andi	37	1.040.000	0.3	0.04	0.04
05	Budi	42	950.000	0	0.2	0
06	Amir	39	1.600.000	0.1	0.6	0.1
07	Rian	37	1.250.000	0.3	0.25	0.25
08	Kiki	32	550.000	0.8	0	0
09	Alda	35	735.000	0.5	0	0
10	Yoga	35	860.000	1	0	0

Dari tabel diatas didapatkan nilai bobot tertinggi derajat keanggotaan nilai dari karyawan yang umur muda dan gaji tinggi yaitu sari dengan bobot nilai 0.4, bobot tersebut didapatkan dengan membandingkan nilai minimum dari kedua fungsi keanggotaan kemudian, dicari nilai yang hasilnya tidak 0.

Tabel 2.8 karyawan yang masih muda tapi memiliki gaji yang tinggi

NIP	Nama	Umur (th)	Gaji/(bl)	Derajat Keanggotaan $\mu[z]$		
				Muda	Tinggi	Muda & tinggi
03	Sari	36	1.500.000	0.4	0.5	0.4

2.6 Penelitian Sebelumnya

Berikut Beberapa buku yang digunakan sebagai referensi pembelajaran “*Artificial Intetelegency*”, Graha Ilmu, disini didapatkan beberapa contoh kasus yang hampir sama dengan permasalahan yang dihadapi,dan juga beberapa artikel dari internet sebagai bahan wacana antara lain :

1. “*Decision Support Untuk Pembelian Mobil Dengan Menggunakan Fuzzy Database Model Tahani*” Oleh Didin Rosyadi Jurusan Teknik Informatika Muhammadiyah Gresik, email masdi2n@yahoo.com, membahas tentang suatu *system* pendukung keputusan yang digunakan dalam penganalisisan data pada proses pemilihan pembelian mobil, dengan penggunaan basis data yang biasa seseorang dapat menangani data-data yang bersifat pasti, deterministik dan presisi. Namun pada kenyatannya seringkali dibutuhkan adanya penanganan pada data-data yang bersifat samar pada sistem basis data. Maka untuk mengatasi masalah tersebut dapat digunakan konsep logika *fuzzy*. Penelitian ini akan mengimplementasikan konsep logika *fuzzy Model Tahani* ke dalam basis data, atau biasa disebut *Fuzzy Database Model Tahani*. Artinya, sistem basis data yang menerapkan konsep *fuzzy Model Tahani* sehingga dapat menangani data-data yang bernilai *fuzzy*. Masalah yang akan diselesaikan adalah proses perekomendasi mobil yang paling sesuai bagi pengguna (calon pembeli mobil). Mobil yang direkomendasikan adalah mobil yang memiliki nilai *fire strength* atau tingkat kesesuaian dengan kriteria pilihan diatas angka 0 (nol) sampai dengan angka 1 (satu). Dengan adanya penelitian ini diharapkan dapat membantu para calon pembeli mobil dalam menentukan mobil yang paling sesuai dengan kriteria pilihannya.
2. “*SPK Menggunakan Basis Data Fuzzy Tahani Untuk Pemilihan Telepon Seluler*”, Oleh Mardia, Universitas Komputer Indonesia

(Unikom), Bandung, membahas tentang pembuatan sistem pendukung keputusan menggunakan basis data *fuzzy* model tahani untuk membantu pemilihan telepon seluler dengan menggunakan beberapa yang berhubungan dengan data yang akan digunakan, kriteria yaitu harga, berat, *talktime*, kamera, panjang, lebar dan tinggi ponsel. Pemilihan ponsel dilakukan dengan menentukan *query* tertentu yang hasilnya berupa telpon seluler yang direkomendasikan. Dengan nilai rekomendasi berkisar antara nilai 0 dan 1 dimana nilai 1 menunjukkan nilai penuh dan mendekati nilai 0 berarti semakin tidak direkomendasikan, dengan adanya *system* pendukung keputusan yang dibuat benar-benar dapat memeberikan rekomendasi yang tepat.