

# TEORI, PRAKTEK & IMPLEMENTASI LOGIKA FUZZY

Ahmad  
Nurjaya



**eureka**  
media aksara

PENERBIT CV. EUREKA MEDIA AKSARA

## **TEORI, PRAKTEK & IMPLEMENTASI LOGIKA FUZZY**

**Penulis** : Ahmad  
Nurjaya

**Desain Sampul** : Ardyan Arya Hayuwaskita

**Tata Letak** : Uli Mas'uliyah Indarwati

**ISBN** :

Diterbitkan oleh : **EUREKA MEDIA AKSARA, JANUARI 2024**  
**ANGGOTA IKAPI JAWA TENGAH**  
**NO. 225/JTE/2021**

### **Redaksi:**

Jalan Banjaran, Desa Banjaran RT 20 RW 10 Kecamatan Bojongsari  
Kabupaten Purbalingga Telp. 0858-5343-1992  
Surel : eurekamediaaksara@gmail.com  
Cetakan Pertama : 2023

### **All right reserved**

Hak Cipta dilindungi undang-undang  
Dilarang memperbanyak atau memindahkan sebagian atau seluruh  
isi buku ini dalam bentuk apapun dan dengan cara apapun,  
termasuk memfotokopi, merekam, atau dengan teknik perekaman  
lainnya tanpa seizin tertulis dari penerbit.

## KATA PENGANTAR

Puji syukur alhamdulillah penulis panjatkan kehadirat Allah SWT atas berkat, rahmat, taufik dan hidayah-Nya, penyusunan buku ini dapat diselesaikan dengan baik. Penulis menyadari bahwa dalam proses penulisan buku ini banyak mengalami kendala, namun berkat bantuan, bimbingan, kerjasama dari berbagai pihak dan berkah dari Allah SWT sehingga kendala-kendala yang dihadapi tersebut dapat diatasi.

Buku ini saya dedikasikan untuk istriku fitriah dan anak-anaku kanaya, elzio dan adreena yang telah memberikan dukungan secara moril, material, arahan, dan selalu mendoakan keberhasilan dan kemudahan selama penyusunan buku ini.

Akhirnya, dengan segala kerendahan hati penulis menyadari masih banyak terdapat kekurangan-kekurangan, sehingga penulis mengharapkan adanya saran dan kritik yang bersifat membangun demi kesempurnaan buku ini.

Pamulang, 2024

## DAFTAR ISI

<b>KATA PENGANTAR.....</b>	<b>iii</b>
<b>DAFTAR ISI.....</b>	<b>iv</b>
<b>DAFTAR GAMBAR.....</b>	<b>vi</b>
<b>DAFTAR TABEL .....</b>	<b>viii</b>
<b>BAB 1 LOGIKA FUZZY.....</b>	<b>1</b>
A. Pengenalan Logika Fuzzy.....	1
B. Contoh Penerapan logika Fuzzy .....	3
C. Himpunan Fuzzy (fuzzy Set) .....	4
D. Fungsi Keanggotaan (Membership Function) .....	5
E. Operator Dasar Himpunan Fuzzy .....	10
F. IF-THEN Rules (Konsep Dasar Aturan Fuzzy) .....	13
G. Tahapan Fuzzy .....	18
<b>BAB 2 PEMROGRAMAN MATLAB.....</b>	<b>24</b>
A. Pengenalan Matlab.....	24
B. Fungsi dan Perintah Dasar Matlab .....	27
C. Command Window .....	34
D. M-Files.....	38
E. Control Flow .....	48
F. Graphical User Interface Builder (GUIDE) MATLAB.....	64
G. Fuzzy Logic Toolbox .....	86
<b>BAB 3 MySQL.....</b>	<b>105</b>
A. Pengenalan MySQL.....	105
B. Kelebihan MySQL .....	108
C. Kekurangan MySQL .....	110
D. Structure Query Language (SQL) .....	110
<b>BAB 4 STUDI KASUS: MEMBANGUN APLIKASI     BERBASIS AI .....</b>	<b>113</b>
A. Pengantar .....	113
B. Karyawan.....	117
C. Kinerja Karyawan.....	117
D. Komponen Penilaian Kinerja Karyawan.....	119
E. Desain dan Rancangan Sistem Fuzzy .....	119
F. Desain dan Rancangan Fungsi Keanggotaan.....	120
G. Desain dan Rancangan Rule System Fuzzy .....	123

H. Membangun Model Sistem Fuzzy.....	138
I. Desain dan Rancangan Aplikasi GUI.....	143
J. Membangun Aplikasi Penilaian Kinerja.....	149
<b>DAFTAR PUSTAKA .....</b>	<b>172</b>
<b>TENTANG PENULIS .....</b>	<b>175</b>

## DAFTAR GAMBAR

<b>Gambar 1.1</b> Contoh Pemetaan Input dan Output .....	2
<b>Gambar 1.2</b> Konsep Umum kronologi pembangunan FIS .....	3
<b>Gambar 1.3</b> Representasi Kurva Linier Naik.....	6
<b>Gambar 1.4</b> Representasi Kurva Linier Turun .....	7
<b>Gambar 1.5</b> Representasi kurva Segitiga .....	8
<b>Gambar 1.6</b> Representasi Kurva Trapesium.....	9
<b>Gambar 1.7</b> Representasi Kurva Bahu.....	10
<b>Gambar 1.8</b> Tabel Kebenaran.....	10
<b>Gambar 1.9</b> Operator Dasar Himpunan Fuzzy .....	11
<b>Gambar 1.10</b> Simulasi Fungsi Operator .....	12
<b>Gambar 1.11</b> Relasi Antar Himpunan.....	14
<b>Gambar 1.12</b> Nilai Antiseden.....	15
<b>Gambar 1.13</b> Nilai Konsekuen .....	16
<b>Gambar 1.14</b> Kurva Hasil Penalaran Monoton .....	16
<b>Gambar 1.15</b> Skema Fungsi Implikasi.....	18
<b>Gambar 1.16</b> Tahapan Fuzzy .....	19
<b>Gambar 2.1</b> Command Window .....	35
<b>Gambar 2.2</b> Tampilan Grafik Sinus .....	36
<b>Gambar 2.3</b> Contoh histogram.....	38
<b>Gambar 2.4</b> Tampilan Editor Matlab.....	39
<b>Gambar 2.5</b> Tampilan Hasil Latihan 1 .....	41
<b>Gambar 2.6</b> Tampilan Hasil Latihan 2 .....	42
<b>Gambar 2.7</b> Tampilan Hasil Latihan 3 .....	44
<b>Gambar 2.8</b> Tampilan Hasil Latihan 4 .....	45
<b>Gambar 2.9</b> Tampilan Hasil Latihan 5 .....	46
<b>Gambar 2.10</b> Tampilan Hasil Latihan 7.....	48
<b>Gambar 2.11</b> Tampilan Hasil Fungsi If .....	50
<b>Gambar 2.12</b> Tampilan Hasil fungsi if-else 1.....	51
<b>Gambar 2.13</b> Tampilan Hasil fungsi if-else 2.....	52
<b>Gambar 2.14</b> Tampilan Latihan For.....	54
<b>Gambar 2.15</b> Tampilan Latihan For-End .....	55
<b>Gambar 2.16</b> Tampilan Latihan ParFor.....	57
<b>Gambar 2.17</b> Tampilan Latihan Switch-Case .....	59
<b>Gambar 2.18</b> Tampilan Latihan Switch-Case-Otherwise.....	60

<b>Gambar 2.19</b>	STampilan Latihan While.....	63
<b>Gambar 2.20</b>	Tampilan Fungsi Grafik.....	64
<b>Gambar 2.21</b>	Struktur File Matlab.....	65
<b>Gambar 2.22</b>	Jendela GUIDE Quick Start.....	66
<b>Gambar 2.23</b>	Lingkungan Komponen GUIDE .....	67
<b>Gambar 2.24</b>	Rancangan Aplikasi Matematika Sederhana .....	70
<b>Gambar 2.25</b>	Tampilan Aplikaasi MTK sederhana.....	75
<b>Gambar 2.26</b>	Rancangan Aplikasi Nilai Mahasiswa .....	75
<b>Gambar 2.27</b>	Tampilan Aplikasi Nilai Mahasiswa .....	79
<b>Gambar 2.28</b>	Rancangan Aplikasi Trigonometri .....	80
<b>Gambar 2.29</b>	Tampilan Aplikasi Trigonometri .....	86
<b>Gambar 2.30</b>	Tampilan Fuzzy Logic Toolbox .....	87
<b>Gambar 2.31</b>	Rancangan Sistem Fuzzy .....	89
<b>Gambar 2.32</b>	Tampilan Aplikasi Deteksi Banjir .....	104
<b>Gambar 3.1</b>	Logo Mysql.....	106
<b>Gambar 4.1</b>	Fungsi Keanggotaan Variabel Absensi .....	120
<b>Gambar 4.2</b>	Fungsi Keanggotaan Variabel Prestasi.....	121
<b>Gambar 4.3</b>	Fungsi Keanggotaan Variabel Disiplin .....	122
<b>Gambar 4.4</b>	Fungsi Keanggotaan Variabel Loyalitas .....	122
<b>Gambar 4.5</b>	Fungsi Keanggotaan Variabel Status Kinerja .....	123
<b>Gambar 4.6</b>	Tampilan Output Model Fuzzzy .....	143
<b>Gambar 4.7</b>	Struktur Folder Aplikasi Penilaian Kinerja .....	150
<b>Gambar 4.8</b>	Tampilan Fungsi Koneksi Database.....	152
<b>Gambar 4.9</b>	Tampilan Laporan Data Karyawan.....	153
<b>Gambar 4.10</b>	Tampilan Laporan Hasil Penilaian.....	155
<b>Gambar 4.11</b>	Tampilan Form Login .....	159
<b>Gambar 4.12</b>	Tampilan Form Utama .....	161
<b>Gambar 4.13</b>	Tampilan Form Data Karyawan .....	166
<b>Gambar 4.14</b>	Tampilan Awal Aplikasi Penilaian Kinerja .....	171

## DAFTAR TABEL

<b>Tabel 2.1</b> Perintah Umum .....	27
<b>Tabel 2.2</b> Perintah Input/Output dan Pemformatan.....	29
<b>Tabel 2.3</b> Perintah Vector, Matrix and Array .....	30
<b>Tabel 2.4</b> Perintah Plotting.....	31
<b>Tabel 2.5</b> Programming.....	33
<b>Tabel 4.1</b> Rancangan Rule Base .....	124
<b>Tabel 4.2</b> Spesifikasi tabel Login.....	144
<b>Tabel 4.3</b> Spesifikasi Tabel Karyawan.....	144
<b>Tabel 4.4</b> Spesifikasi Tabel Penilaian.....	144
<b>Tabel 4.5</b> Rancangan AntarMuka .....	145



# BAB 1

# LOGIKA FUZZY

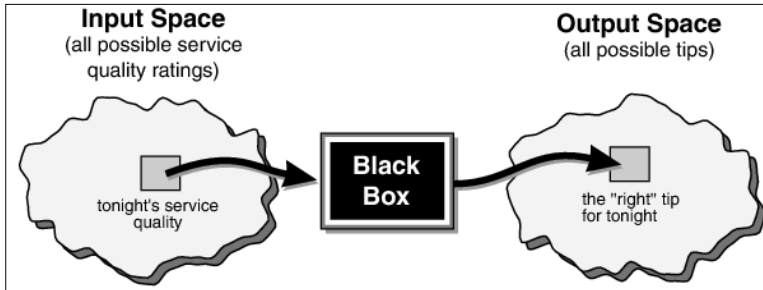
## A. Pengenalan Logika Fuzzy

Kata fuzzy merupakan kata sifat yang berarti kabur, tidak jelas. Fuzziness atau kekaburan atau ketidakjelasan selalu meliputi keseharian manusia. Logika fuzzy dikatakan sebagai logika baru yang lama, sebab ilmu tentang Logika fuzzy modern baru ditemukan beberapa tahun yang lalu. Sebenarnya konsep tentang Logika fuzzy itu sendiri sudah ada sejak lama (Karmilasari, 2008).

Teori *fuzzy* pertama kali diperkenalkan oleh Dr. Lotfi Zadeh pada tahun 1965 dari universitas California (Vasanti, 2017), untuk menyusun konsep kualitatif yang tidak punya batasan-batasan tepat, sebagai contoh tidak ada nilai yang jelas atau pasti yang mewakili batasan antara normal dan rendah, atau tinggi dan normal. Secara bahasa dalam kamus *Oxford* istilah *fuzzy* didefinisikan sebagai *blurred* (kabur atau remang-remang), *indistinct* (tidak jelas), *imprecisely defined* (didefinisikan secara tidak presisi), *confused* (membingungkan), *vague* (tidak jelas) (Tanaamah, Advendi, & Pakereng, 2012), istilah ini tidak mengacu pada suatu sistem yang tidak jelas/kabur/remang-remang definisinya, cara kerjanya atau detesisnya, tetapi sebaliknya yang dimaksud dengan sistem *fuzzy* adalah sebuah sistem yang dibangun dengan definisi, cara kerja dan detesis yang jelas berdasarkan kepada teori Logika *fuzzy* (Anbarasu, Jenitha, & Yulit, 2015).

Logika *fuzzy* adalah sebuah metode yang digunakan untuk menjelaskan ketidakpastian dengan cara pemetaan input

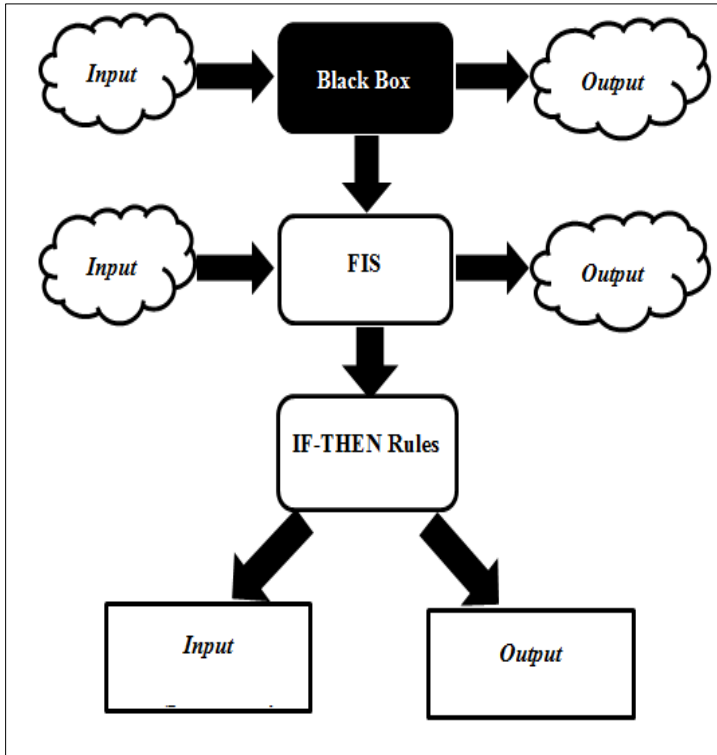
kedalam ruang output. Logika *fuzzy* merupakan suatu cara yang cocok untuk melakukan pemetaan atau *mapping* hubungan *input* dan *output* dari suatu sistem berdasarkan data *input* dan *output*.



**Gambar 1.1** Contoh Pemetaan Input dan Output

Pada umumnya dalam kehidupan sehari-hari kita sering menjumpai ketidakjelasan, seperti keambiguan, Keacakan, Ketidaktepatan, Ketidakjelasan akibat informasi yang tidak lengkap, dan kekaburan semantik, dari kelima kelompok ketidakjelasan tersebut, dapat dikatakan bahwa pembahasan Logika *fuzzy* berada pada kekaburan semantik (Kusumadewi, 2003), kekaburan semantik pasti ada dalam kehidupan manusia. Bahkan kita sering mengambil keputusan dari kondisi kekaburan semantik.

Motivasi utama dari konsep Logika *fuzzy* adalah memetakan sebuah ruang *input* kedalam ruang *output* dengan menggunakan *IF-THEN rules*. Pemetaan ini dilakukan dalam suatu *fuzzy Inferensi system (FIS)*. Urutan *rule* bisa sembarang, *FIS* mengevaluasi semua *rule* secara simultan untuk menghasilkan kesimpulan, oleh karena itu semua *rule* sebelum kita membangun *FIS* yang akan digunakan untuk menginterpretasikan semua *rule* tersebut (Shaout & Yousif, 2014).



**Gambar 1.2** Konsep Umum kronologi pembangunan FIS

### **B. Contoh Penerapan logika Fuzzy**

Kelebihan dari fuzzy yang bersifat fleksibel dan mampu mentolerir ketidakpastian serta mudah di mengerti menjadikan logika fuzzy sampai saat ini banyak digunakan dan di terapkan kedalam berbagai bidang, antara lain:

1. Sistem Kendali; di terapkan padakendali mesin cuci, kendali AC, sistem pengereman otomatis pada mobil, dan lain sebagainya.
2. Sistem diagnosa; logika fuzzy juga banyak diterapkan pada dunia kedokteran untuk sistem diagnosa berbagai macam penyakit, seperti TBC Kanker, dan lain-lain
3. Sistem prediksi; dapat juga diterapkan untuk sistem peramalan atau prediksi seperti prediksi cuaca, prediksi harga saham ataupun lainnya

4. Sistem seleksi; fuzzy juga dapat digunakan untuk melakukan proses seleksi, seperti untuk sistem seleksi pemberin beasiswa, seleksi penerimaan karyawan dan lain sebagainya.
5. Sistem deteksi; fuzzy juga banyak diterapkan kedalam sistem deteksi, seperti deteksi kebakaran, sistem deteksi gempa bumi, sistem deteksi tsunami dan masih banyak lagi.

### C. Himpunan Fuzzy (fuzzy Set)

Pada himpunan tegas (*crisp*), nilai keanggotaan suatu item  $x$  dalam suatu himpunan  $A$ , yang sering ditulis dengan  $\mu_A[x]$  (Kusumadewi, 2003), memiliki 2 kemungkinan yaitu:

1. Satu (1), yang berarti bahwa suatu item menjadi anggota dalam suatu himpunan
2. Nol (0), yang berarti bahwa suatu item tidak menjadi anggota dalam suatu himpunan.

Disini bisa dikatakan bahwa pemakaian himpunan *crisp* untuk menyatakan umur sangat tidak adil (Nurjaya, 2022), adanya perubahan kecil saja pada suatu nilai mengakibatkan perbedaan kategori yang sangat signifikan. Himpunan *fuzzy* digunakan untuk mengantisipasi hal tersebut. Seseorang dapat masuk dalam 2 himpunan yang berbeda. Seberapa besar eksistensinya dalam himpunan tersebut dapat dilihat pada nilai keanggotaannya.

Kalau pada himpunan *crisp*, nilai keanggotaan hanya ada 2 kemungkinan yaitu 0 dan 1, pada himpunan *fuzzy* nilai keanggotaan terletak pada rentang 0 sampai 1. Apabila  $x$  memiliki nilai keanggotaan *fuzzy*  $\mu_A[x] = 0$  berarti  $x$  tidak menjadi anggota himpunan  $A$ , demikian pula apabila  $x$  memiliki nilai keanggotaan *fuzzy*  $\mu_A[x] = 1$  berarti  $x$  menjadi anggota penuh pada himpunan  $A$ .

Terkadang kemiripan antara keanggotaan *fuzzy* dengan probabilitas menimbulkan kerancuan. Keduanya memiliki nilai pada interval  $[0,1]$ , namun interpretasi nilainya sangat berbeda antara kedua kasus tersebut. Keanggotaan *fuzzy* memberikan suatu ukuran terhadap pendapat dan keputusan (Anbarasu, Jenitha, & Yulit, 2015), sedangkan probabilitas mengindikasikan

proporsi terhadap keseringan suatu hasil bernilai benar dalam jangka panjang. Misalnya, jika nilai keanggotaan suatu himpunan *fuzzy* muda adalah 0,9; maka tidak perlu dipermasalahkan berapa seringnya nilai itu diulang secara individual untuk mengharapkan suatu hasil yang hampir pasti muda. Di lain pihak, nilai probabilitas 0,9 muda berarti 10% dari himpunan tersebut diharapkan tidak muda.

Himpunan *fuzzy* memiliki 2 atribut (Vasanti, 2017) , yaitu:

1. *Linguistic*, yaitu penamaan suatu grup yang mewakili suatu keadaan atau kondisi tertentu dengan menggunakan bahasa alami
2. *Numeris*, yaitu suatu nilai (angka) yang menunjukkan ukuran dari suatu variabel seperti: 40, 30, 29.

Ada beberapa hal yang perlu diketahui dalam memahami sistem *fuzzy* (Kusumadewi, 2006), yaitu:

1. Variabel *fuzzy* merupakan variabel yang hendak dibahas dalam suatu sistem *fuzzy*.
2. Himpunan *fuzzy* merupakan suatu grup yang mewakili suatu kondisi atau keadaan tertentu dalam suatu variabel *fuzzy*.
3. Semesta pembicaraan adalah keseluruhan nilai yang diperbolehkan untuk dioperasikan dalam suatu Variabel *fuzzy*. Semesta pembicaraan merupakan himpunan bilangan *real* yang senantiasa naik (bertambah) secara monoton dari kiri ke kanan.
4. Domain adalah keseluruhan nilai yang diijinkan dalam semesta pembicaraan dan boleh dioperasikan dalam suatu himpunan *fuzzy*. Seperti halnya semesta pembicaraan, domain merupakan himpunan bilangan *real* yang senantiasa naik (bertambah) secara *monoton* dari kiri ke kanan.

#### **D. Fungsi Keanggotaan (Membership Function)**

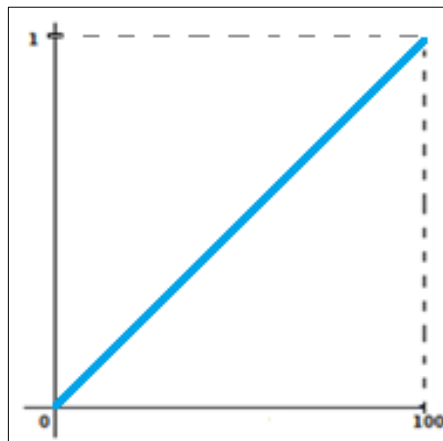
Fungsi keanggotaan adalah suatu kurva yang menunjukkan pemetaan titik-titik ruang *input* ke dalam ruang output dengan cara ruang input dipetakan kedalam bentuk nilai keanggotaannya (sering juga disebut dengan derajat keanggotaan) yang memiliki interval antara 0 sampai 1. Salah

satu cara yang dapat digunakan untuk mendapatkan nilai keanggotaan adalah dengan melalui pendekatan fungsi. Didalam fungsi keanggotaan terdapat banyak representasi kurva, seperti kurva linier, kurva segitiga, kurva s, kurva lonceng dan masih banyak lagi, tetapi ada beberapa fungsi yang biasa digunakan (Tanaamah, Advendi, & Pakereng, 2012), antara lain:

#### 1. Representasi kurva linier

Pada Representasi kurva linier, pemetaan *input* ke derajat keanggotaannya digambarkan sebagai suatu garis lurus (Shaout & Yousif, 2014), bentuk ini paling sederhana dan menjadi pilihan yang baik untuk mendekati suatu konsep yang kurang jelas.

Ada dua keadaan himpunan *fuzzy* yang *linier*, pertama, kenaikan himpunan dimulai pada nilai domain yang memiliki derajat keanggotaan nol [0] bergerak kekanan menuju ke nilai *domain* yang memiliki derajat keanggotaan lebih tinggi atau sering disebut dengan *Representasi* kurva *linier* naik.

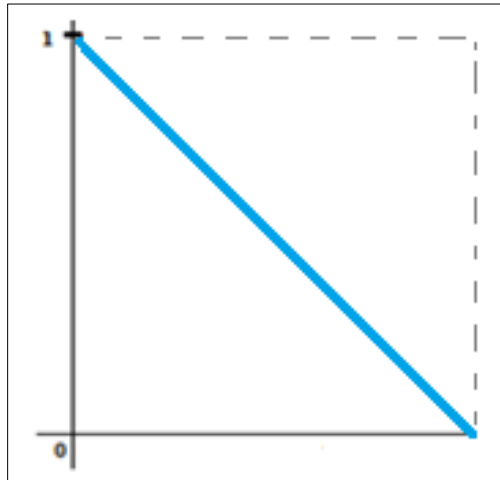


**Gambar 1.3** Representasi Kurva Linier Naik

Fungsi keanggotaan

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

Kedua, merupakan kebalikan dari yang pertama (Kusumadewi, 2003), 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 atau sering disebut dengan kurva *linier* turun.



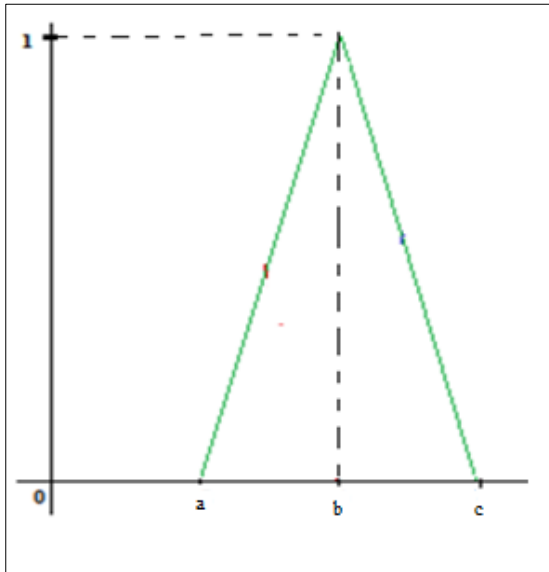
**Gambar 1.4** Representasi Kurva Linier Turun

Fungsi keanggotaan:

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

## 2. Representasi kurva Segitiga

Kurva segitiga pada dasarnya merupakan gabungan antara 2 garis (linier).



**Gambar 1.5** Representasi kurva Segitiga

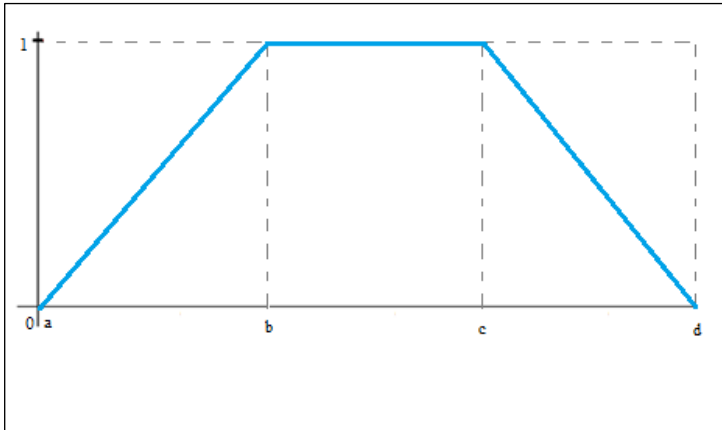
Fungsi keanggotaan:

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

## 3. Representasi Kurva Trapezium

Kurva trapezium juga pada dasarnya merupakan gabungan dari kurva linier naik dan linier turun, (Nurjaya, 2022) tetapi dengan wilayah crisp atau tegas di yang berada diantaranya, adapun bentuk kurva trapezium dapat dilihat pada gambar berikut ini.





**Gambar 1.6** Representasi Kurva Trapesium

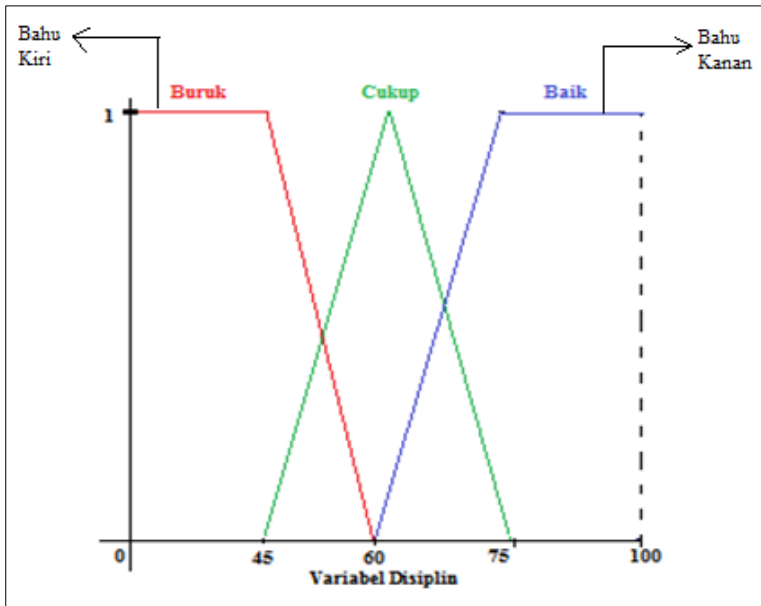
Fungsi keanggotaan:

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

#### 4. Representasi Kurva Bahu

Kurva bahu atau disebut juga kurva setengah trapezium, pada kurva ini pada dasarnya terdapat dua jenis, yaitu kurva bahu kiri dan krva bahu kanan, dimana pada kurva ini fungsi keanggotaan kurva bahu kiri memiliki kesamaan dengan kurva linier turun, sedangkan untuk kurva bahu kanan memiliki fungsi keanggotaan kurva linier naik.

Kurva jenis ini digunakan untuk mengakhiri variabel dari sebuah daerah *fuzzy*, adapun bentuk kurva bahu dapat dilihat pada gambar berikut ini.



**Gambar 1.7** Representasi Kurva Bahu

#### E. Operator Dasar Himpunan Fuzzy

Hal yang paling penting untuk disadari tentang penalaran logika fuzzy adalah kenyataan bahwa ini adalah superset dari standar logika Boolean. Dengan kata lain, jika kita menjaga nilai fuzzy pada titik ekstrem 1 (sepenuhnya benar), dan 0 (sepenuhnya salah), operasi logika standar akan berlaku. Sebagai contoh, perhatikan tabel kebenaran standar berikut.

A	B	A and B
0	0	0
0	1	0
1	0	0
1	1	1
<b>AND</b>		

A	B	A or B
0	0	0
0	1	1
1	0	1
1	1	1
<b>OR</b>		

A	not A
0	1
1	0
<b>NOT</b>	

**Gambar 1.8** Tabel Kebenaran

Mengingat bahwa, dalam logika fuzzy, kebenaran suatu pernyataan tergantung pada derajatnya, maka nilai masukan dapat berupa bilangan real antara 0 dan 1. Fungsi apa yang dapat mempertahankan hasil dari operator AND pada tabel kebenaran (misalnya) dan juga mencakup semua bilangan real antara 0 dan 1?

Salah satu jawabannya adalah operasi min . Artinya, selesaikan pernyataan  $A \text{ AND } B$  , di mana A dan B dibatasi pada rentang  $(0,1)$ , dengan menggunakan fungsi  $\min(A, B)$  . Dengan menggunakan alasan yang sama, kita dapat mengganti operasi OR dengan fungsi max, sehingga  $A \text{ OR } B$  menjadi ekuivalen dengan  $\max(A, B)$  . Akhirnya, operasi NOT A menjadi ekuivalen dengan operasi tersebut  $1 - A$  Tabel kebenaran sebelumnya tidak berubah sama sekali dengan substitusi ini.

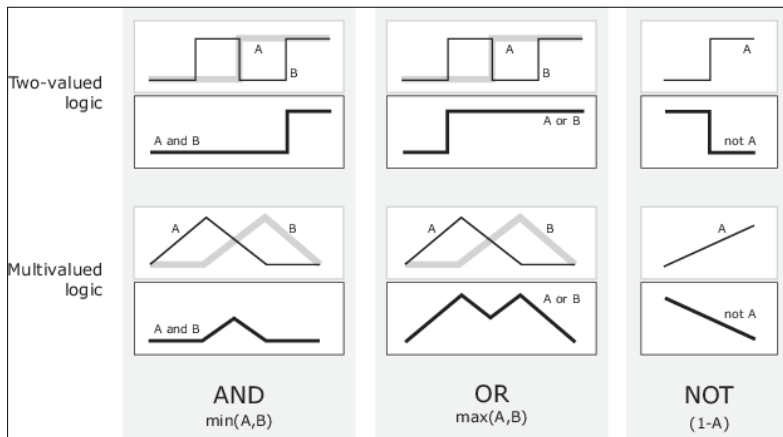
A	B	$\min(A,B)$	A	B	$\max(A,B)$	A	$1 - A$
0	0	0	0	0	0	0	1
0	1	0	0	1	1	1	0
1	0	0	1	0	1		
1	1	1	1	1	1		
<b>AND</b>			<b>OR</b>			<b>NOT</b>	

Gambar 1.9 Operator Dasar Himpunan Fuzzy

Selain itu, karena terdapat fungsi di balik tabel kebenaran dan bukan hanya tabel kebenaran itu sendiri, kini kita dapat mempertimbangkan nilai selain 1 dan 0.

Gambar berikutnya menggunakan grafik untuk menunjukkan informasi yang sama. Pada gambar ini, tabel kebenaran diubah menjadi plot dua himpunan fuzzy yang diterapkan bersama untuk membuat satu himpunan fuzzy. Bagian atas gambar menampilkan plot yang sesuai dengan tabel kebenaran dua nilai sebelumnya, sedangkan bagian bawah gambar menampilkan cara kerja operasi pada

rentang nilai kebenaran *A dan B* yang terus bervariasi sesuai dengan operasi fuzzy yang telah kita tetapkan.



**Gambar 1.10** Simulasi Fungsi Operator

Dengan adanya ketiga fungsi ini, kita dapat menyelesaikan konstruksi apa pun menggunakan himpunan fuzzy dan operasi logika fuzzy AND, OR, dan NOT.

#### 1. Operator AND

Operator ini digunakan untuk mendefinisikan operasi interseksi atau perpotongan dua himpunan fuzzy atau yang dikenal sebagai persimpangan atau konjungsi fuzzy (AND). Hasil dari penggunaan operator ini berupa nilai  $\alpha$  predikat, yang diperoleh dengan mengambil nilai derajat keanggotaan terkecil dari elemen pada himpunan yang terkait.

$$\mu_{A \cap B}(X) = \min(\mu_A[X], \mu_B[X])$$

#### 2. Operator OR

Operator ini berhubungan dengan operasi union atau penyatuan atau disjungsi fuzzy (OR). Hasil dari penggunaan operator ini berupa nilai  $\alpha$  predikat, yang diperoleh dengan mengambil nilai derajat keanggotaan terbesar dari elemen pada himpunan yang terkait.

$$\mu_{A \cup B}(X) = \max(\mu_A[X], \mu_B[X])$$

### 3. Operator NOT

Operator ini berhubungan dengan operasi komplemen fuzzy (NOT). Hasil dari penggunaan operator ini berupa nilai  $\alpha$  predikat, yang diperoleh dengan mengurangi nilai derajat keanggotaan dari elemen pada himpunan yang terkait dengan nilai 1.

$$\mu_{A'} = 1 - \mu_A[X]$$

## F. IF-THEN Rules (Konsep Dasar Aturan Fuzzy)

Himpunan fuzzy dan operator fuzzy merupakan subjek dan kata kerja logika fuzzy. Pernyataan aturan if then (jika-maka) ini digunakan untuk merumuskan pernyataan kondisional yang terdiri dari logika fuzzy.

Adapun bentuk dasar dari aturan fuzzy tersebut, adalah

*If X is A then Y is B*

*Atau*

*If X1 is A ; X2 is C then Y is B*

Dimana A dan B adalah nilai linguistik yang ditentukan oleh himpunan fuzzy pada rentang (semesta pembicaraan) X dan Y, masing-masing. Bagian *if* dari aturan tersebut yang merupakan input dari sistem fuzzy (*X is A*) disebut *anteseden* atau premis, bagian *;* dari aturan tersebut yang merupakan operator logika (AND, OR, NOT), sedangkan bagian *then* dari aturan tersebut yang merupakan output dari sistem fuzzy (*Y is B*) disebut *konsekuensi* atau kesimpulan. Dalam mencari atau menentukan nilai dari konsekuensi dapat dilakukan dengan 2 teknik yaitu penalaran monoton dan juga fungsi implikasi:

### 1. Fungsi Penalaran Monoton

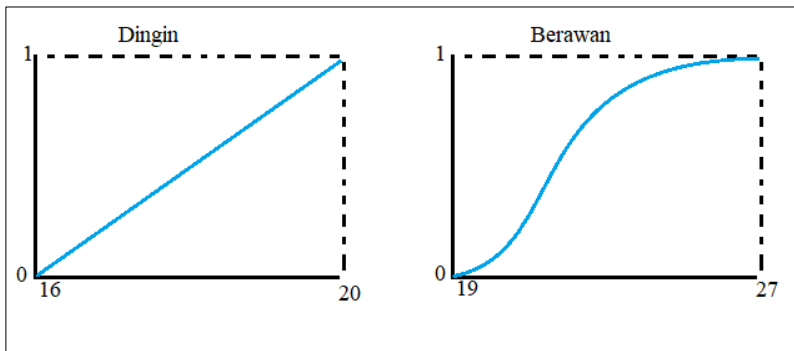
Teknik penalaran monoton memang sudah jarang digunakan, akan tetapi teknik ini merupakan dasar dari teknik implikasi sistem fuzzy.

Contoh bentuk aturan:

*If Suhu is Dingin then Cuaca is Berawan*

Dari aturan tersebut maka daerah fuzzy direlasikan dengan implikasi sederhana, dimana sistem fuzzy dapat

berjalan tanpa komposisi dan dekomposisi fuzzy, nilai output dapat disetimasikan secara langsung dari nilai derajat keanggotaan yang berhubungan dengan antisedennya (Penalaran monoton). Himpunan **Dingin** direpresentasikan dengan derajat keanggotaan (sebagai angka) antara 0 dan 1, sehingga antesedennya adalah interpretasi yang mengembalikan nilai satu angka antara 0 dan 1. Sebaliknya, **Berawan** direpresentasikan sebagai himpunan fuzzy, sehingga konsekuensinya adalah penugasan yang menugaskan seluruh himpunan fuzzy B ke variabel keluaran atau output Y. Secara umum, masukan pada aturan if-then adalah nilai saat ini untuk variabel masukan atau input (dalam hal ini, *suhu*) dan keluarannya atau output berupa keseluruhan himpunan fuzzy (dalam hal ini, *Cerah*), himpunan ini nantinya akan *didefuzzifikasi* sehingga akan menghasilkan satu nilai pada output. Dengan kurva dasar seperti gambar dibawah ini



**Gambar 1.11** Relasi Antar Himpunan

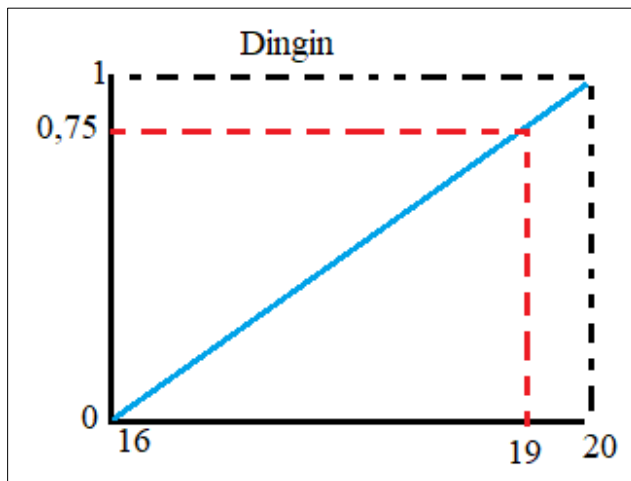
Maka langkah penyelesaiannya adalah,

1. Tentukan nilai derajat kenggotaan pada variabel suhu didalam domain himpunan dingin
2. Pada variabel output petakan dengan menaarik garis lurus pada domain himpunan berawan

Dengan relasi kedua himpunan tersebut yang telah diekspresikan dengan aturan sederhana diatas, maka dengan fungsi penalaran monoton, dapat diformulasikan

$$Yb = F(\mu_A[X], Db)$$

Pada langkah 1, menentukan derajat keanggotaan himpunan dingin pada variabel suhu, jika input pada variabel suhu adalah 19°C, maka diperoleh derajat keanggotaan berikut ini



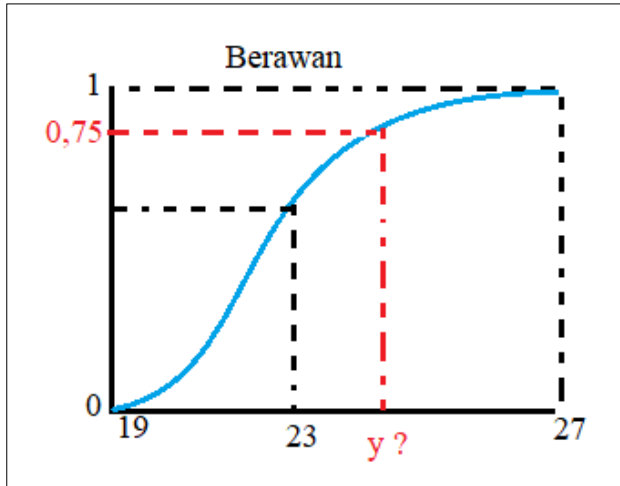
**Gambar 1.12** Nilai Antiseden

dengan formula

$$\begin{aligned}\mu_{dingin} &= (19-16)/(20-16) \\ &= 3/4 \\ &= 0,75\end{aligned}$$

Pada langkah 2, dilakukan pemetaan didalam konsekuensi (variabel output) pada domain himpunan berawan, dengan formula:

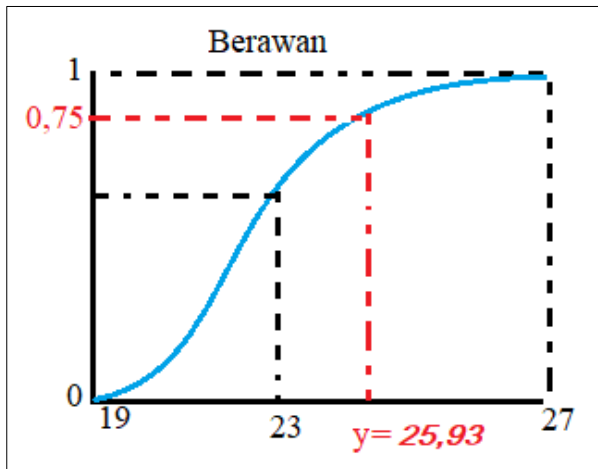
$$\mu_{berawan}[y] = S(y; 19; 23; 27) = 0,75$$



**Gambar 1.13** Nilai Konsekuen

Pada gambar diatas, diketahui bahwa nilai  $0,75 > 0,5$ , maka letak kemungkinan nilai outputnya berada pada rentan nilai 23-27, didapatkan hasil dengan menggunakan formula:

$$\begin{aligned}
 \mu_{dingin} &= (y-19)^2 / (27-19)^2 \\
 &= (y-19)^2 / 64 \\
 &= 0,75 \\
 &= 25,93
 \end{aligned}$$



**Gambar 1.14** Kurva Hasil Penalaran Monoton



Dengan fungsi penalaran monoton, dari rule sederhana di atas didapatkan hasil sebesar 25,93°C untuk himpunan berawan pada variabel cuaca.

## 2. Fungsi Implikasi

Selain bentuk aturan if-then diatas, dimana antisedennya hanya melibatkan satu variabel input saja, bentuk aturan if-then bisa memiliki dua variabel input atau bahkan lebih sebagai antiseden, seperti rule if-then berikut ini.

*If Suhu is Panas And Kecepatan Angin is Tinggi And Kelembaban udara is tinggi then Cuaca is ... ....*

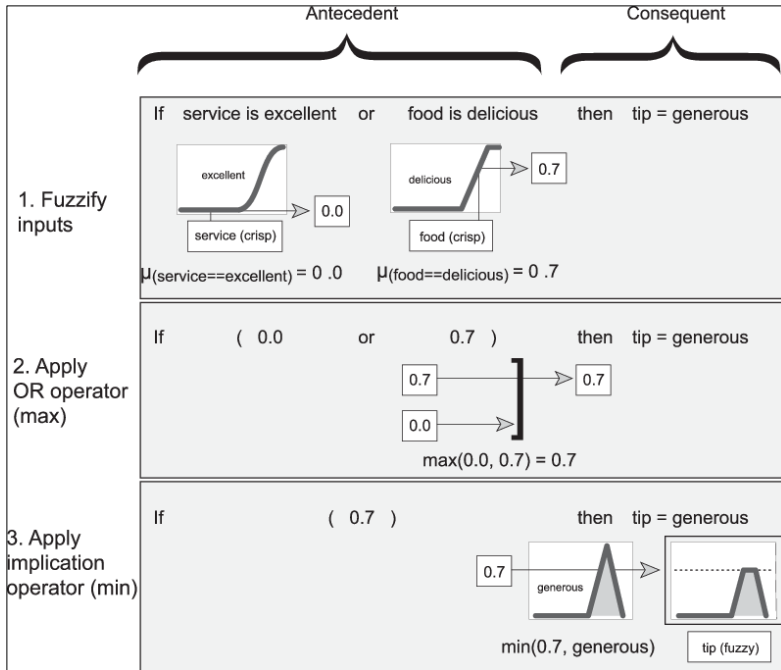
Dalam hal ini, seluruh bagian anteseden dihitung secara bersamaan dan diselesaikan menjadi satu angka menggunakan operator logika (AND, OR, NOT) yang dijelaskan di bagian sebelumnya.

Atau bentuk aturan if-then bisa saja variabel output atau konsekuensi suatu aturan memiliki banyak variabel, seperti contoh berikut ini.

*If Beban Pakaian is Ringan And Tingkat Kekotoran is Sedang then Waktu Cuci is Cepat And Putaran Rotor Is Sedang*

Dalam hal ini, semua konsekuensi dipengaruhi secara sama oleh hasil dari anteseden. Konsekuensinya menentukan himpunan fuzzy yang ditugaskan ke kedalam variabel output. Fungsi *implikasi* kemudian memodifikasi himpunan fuzzy tersebut ke derajat yang ditentukan oleh anteseden.

Secara umum untuk memodifikasi himpunan fuzzy pada variabel output adalah pemotongan menggunakan fungsi min(di mana himpunan fuzzy terpotong seperti yang ditunjukkan pada gambar berikut) atau penskalaan menggunakan fungsi prod(di mana himpunan fuzzy keluaran diapit).



**Gambar 1.15** Skema Fungsi Implikasi

## G. Tahapan Fuzzy

Dalam sistem Logika *fuzzy* terdapat beberapa tahapan *operasional* yang meliputi fuzzifikasi, inferensi dan defuzzifikasi (Alamsyah & Muna, 2016), untuk lebih jelasnya dapat dilihat pada gambar tahapan fuzzy berikut ini;



**Gambar 1.16** Tahapan Fuzzy

1. *Fuzzyfikasi*

*Fuzzyfikasi* merupakan proses pengubahan nilai *input* yang berada dalam suatu himpunan tegas menjadi nilai *input* yang berada dalam suatu himpunan *fuzzy* (Kusumadewi, 2003). Dalam menentukan apakah suatu elemen merupakan anggota dari suatu himpunan *fuzzy* tidak semudah himpunan tegas (Vasanti, 2017), dalam himpunan tegas kita dapat langsung menentukan keanggotaan elemen. Sedangkan dalam himpunan *fuzzy* kita tidak dapat langsung menentukan keanggotaan elemen. Zadeh mengaitkan himpunan yang memiliki batas tak jelas dengan suatu fungsi yang menyatakan derajat kesesuaian unsur-unsur dalam semestanya melalui konsep yang merupakan syarat keanggotaan himpunan tersebut. Derajat keanggotaan

berupa bilangan riil dalam interval tertutup  $[0,1]$ . Dengan demikian, fungsi keanggotaan dari suatu himpunan *fuzzy* dalam semesta  $X$  adalah pemetaan  $\mu_A$  dari  $X$  ke interval  $[0,1]$ .

$$\mu_A : x \rightarrow [0,1]$$

Nilai fungsi  $\mu_A(x)$  menyatakan derajat/ nilai keanggotaan unsur (elemen)  $x \in X$  dalam himpunan *fuzzy*  $A$ .

- a. Nilai fungsi = 1 menyatakan keanggotaan penuh (sempurna).
- b. Nilai fungsi = 0 menyatakan bukan merupakan anggota himpunan

## 2. Inferensi

Inferensi merupakan proses pemberian aturan sebagai basis pengetahuan seorang ahli. Suatu aturan *fuzzy* dapat dituliskan sebagai:

***IF anteseden THEN konsekuen***

Proses *inferensi* dalam suatu sistem berbasis aturan *fuzzy* akan memperhitungkan semua aturan yang ada di dalam basis pengetahuan, hasil dari proses *inferensi* ini direpresentasikan oleh suatu himpunan *fuzzy* untuk setiap variabel bebas pada konsekuen. Sedangkan derajat keanggotaan untuk setiap variabel tidak bebas menyatakan ukuran kompatibilitas terhadap variabel bebas pada anteseden.

Terdapat tiga metode dalam pemberian aturan kedalam sistem *fuzzy* (*Fuzzy Inference System*) yang dapat digunakan (Anbarasu, Jenitha, & Yulit, 2015), yaitu:

### a. Metode Tsukamoto

Pada metode ini output dari basis aturan *fuzzy* berupa himpunan dari variabel output, dimana hasil dari inferensi setiap aturan berupa nilai tegas (crisp) yang dibuat berdasarkan nilai  $\alpha$ predikat, dan pada tahap akhir (defuzzifikasi) menggunakan rata-rata terbobot. Adapun bentuk dasar rule sistem dengan metode tsukamoto adalah sebagai berikut;

**IF  $x_1$  is  $A_1$  AND  $x_2$  is  $A_2$  AND...AND  $x_n$  is  $A_n$  THEN  $y$  is  $B$**

b. Metode Mamdani

Pada metode ini bentuk aturan sistem sama dengan bentuk aturan sistem pada metode tsukamoto, aturan *fuzzy* didefinisikan sebagai:

**IF**  $x_1$  is  $A_1$  **AND**  $x_2$  is  $A_2$  **AND**...**AND**  $x_n$  is  $A_n$  **THEN**  $y$  is  $B$

dimana  $A_1, A_2, \dots$  And, dan  $B$  adalah nilai-nilai linguistik atau himpunan *fuzzy*, sedangkan pernyataan " $x_1$  is  $A_1$ " menyatakan bahwa  $x_1$  adalah anggota dari himpunan *fuzzy*  $A_1$ .

c. Metode Sugeno

Model ini merupakan varian dari model Mamdani yang dikenal juga sebagai Takagi-Sugeno-Kang (TSK). Aturan *fuzzy* yang digunakan model ini adalah:

**IF**  $x_1$  is  $A_1$ =**AND**  $x_2$  is  $A_2$  **AND**..**AND** is  $A_n$  **THEN**  $y=f(x_1,x_2,\dots,x_n)$

dimana  $f$  bisa merupakan sembarang fungsi dari variabel-variabel *input* yang nilainya berada dalam interval variabel *output*. Umumnya fungsi ini dibatasi dengan menyatakan  $f$  sebagai kombinasi *linier* dari variabel-variabel *input*:

$$F(x_1,x_2,\dots,x_n)=w_0+w_1x_1+w_2x_2+\dots+w_nx_n$$

Dimana  $w_1, w_2, \dots, w_n$  adalah konstanta yang berupa bilangan riil dan merupakan bagian dari spesifikasi aturan *fuzzy*. Bentuk tersebut dikenal sebagai model Sugeno orde satu. Terdapat model lain yaitu Sugeno orde nol dimana:

$$f = (x_1, x_2, \dots, x_n)$$

### 3. Defuzzifikasi

Fuzzifikasi merupakan proses pengubahan nilai output dari bentuk bilangan fuzzy kedalam bentuk bilangan tegas sebagai output dari sistem fuzzy, Terdapat banyak metode defuzzifikasi yang telah berhasil diaplikasikan, diantaranya:

#### a. Centroid Method

Metode ini dikenal juga sebagai *Center of Area (CoA)* atau *Center of Gravity (CoG)*. Rumus yang digunakan adalah:

$$y^* = \frac{\int y \mu_R(y) dy}{\int \mu_R(y) dy}$$

Namun jika nilai  $y$  berupa nilai diskrit, maka dapat digunakan rumus:

$$y^* = \frac{\sum y \mu_R(y)}{\sum \mu_R(y)}$$

#### b. Height Method

Metode ini dikenal juga sebagai metode singleton atau keanggotaan maksimum, karena secara sederhana akan memilih nilai tegas (*crisp*) yang memiliki derajat keanggotaan maksimum.

#### c. First of Maxima/Last of Maxima Method

Metode ini merupakan generalisasi dari *height method* dimana fungsi keanggotaan *output* dapat memiliki lebih dari satu nilai maksimum, oleh karenanya akan dipilih nilai maksimum pertama atau nilai maksimum terakhir, tergantung aplikasi yang dibuat.

#### d. Mean-Max Method

Metode ini dikenal juga sebagai *Middle of Maxima*. Metode ini juga merupakan generalisasi dari *height method*, namun akan memilih nilai tengah dari nilai tegas dengan derajat keanggotaan maksimum, dengan rumus:

$$y^* = \frac{m+M}{2}$$

dimana  $m$  adalah nilai tegas yang terkecil dan  $M$  adalah nilai tegas yang terbesar.

#### e. Weighted Average Method

Metode ini mengambil nilai rata-rata dengan menggunakan pembobotan yang diambil dari nilai derajat keanggotaan, yaitu dengan rumus:

$$y^* = \sum \frac{\mu(y) - y}{\mu(y)}$$

dimana  $y$  adalah nilai tegas dan  $\mu(y)$  adalah derajat keanggotaan dari nilai  $y$  tersebut.

# BAB 2

## PEMROGRAMAN MATLAB

### A. Pengenalan Matlab

MATLAB singkatan dari Matrix Laboratory, merupakan bahasa pemrograman tingkat tinggi yang dikembangkan oleh MathWorks dan dikhususkan untuk komputasi numerik, visualisasi, dan pemrograman (Sugiharto, 2006), dengan memanfaatkan MATLAB, pengguna dapat melakukan analisis data, mengembangkan algoritma, dan membuat model maupun aplikasi bahasa, tools, dan fungsi-fungsi built-in akan memudahkan pengguna untuk mengeksplorasi berbagai pendekatan dan memperoleh solusi dengan lebih cepat dibandingkan apabila menggunakan spreadsheets atau bahasa pemrograman tradisional (Rani, 2013), seperti C/C++ atau Java™.

Beberapa manfaat yang didapatkan dari Matlab antara lain:

1. Perhitungan Matematika
2. Komputasi numeric
3. Simulasi dan pemodelan
4. Visualisasi dan analisis data
5. Pembuatan grafik untuk keperluan sains dan teknik
6. Pengembangan aplikasi, misalnya dengan memanfaatkan GUI.



Matlab merupakan aplikasi yang lengkap dengan toolbox yang dapat dimanfaatkan untuk memecahkan masalah sains dan teknik. Dalam matlab terdapat beberapa bagian penting (Sugiharto, 2006), antara lain:

1. Command window, berfungsi untuk menuliskan perintah singkat pada matlab dan juga untuk menampilkan hasil eksekusi
2. Workspace, merupakan jendela yang berisi tentang informasi pemakaian memori didalam matlab
3. Command History, adalah sebuah jendela yang berisi informasi tentang perintah yang pernah dijalankan sebelumnya
4. Preferences, merupakan fasilitas yang digunakan untuk pengaturan segala fasilitas yang terdapat pada matlab
5. Current Directory, fasilitas yang digunakan untuk menemukan direktori yang aktif pada matlab.
6. Launch pad, merupakan fasilitas yang berfungsi untuk memudahkan akses dari produk-produk matlab seperti demo dan dokumentasi.

Fitur-fitur MATLAB sudah banyak dikembangkan, dan lebih kita kenal dengan nama toolbox (MathWorks, 2011), toolbox ini merupakan kumpulan dari fungsi-fungsi MATLAB (M-files) yang telah dikembangkan ke suatu lingkungan kerja MATLAB untuk memecahkan masalah. Area-area yang sudah bisa dipecahkan dengan toolbox saat ini meliputi pengolahan sinyal, sistem kontrol, neural networks, fuzzy logic, wavelets, dan lain-lain.

Sebagai sebuah sistem, MATLAB tersusun dari 5 bagian utama:

1. Development Environment merupakan sekumpulan perangkat dan fasilitas yang membantu kita untuk menggunakan fungsi-fungsi dan file-file MATLAB, beberapa perangkat ini merupakan sebuah graphical user interfaces (GUI) termasuk didalamnya adalah MATLAB desktop dan Command Window, command history, sebuah editor dan

debugger, dan browsers untuk melihat help, workspace, files, dan search path.

2. MATLAB Mathematical Function Library merupakan sekumpulan algoritma komputasi mulai dari fungsi-fungsi dasar seperti: sum, sin, cos, dan complex arithmetic, sampai dengan fungsi-fungsi yang lebih kompleks seperti matrix invers, matrix eigenvalues, Bessel functions, dan fast Fourier transforms.
3. MATLAB Language merupakan suatu high-level matrix atau array language dengan control flow statements, functions, data structures, input/output, dan fitur-fitur object-oriented programming. Ini memungkinkan bagi kita untuk melakukan kedua hal baik pemrograman dalam lingkup sederhana untuk mendapatkan hasil yang cepat, dan pemrograman dalam lingkup yang lebih besar untuk memperoleh hasil-hasil dan aplikasi yang kompleks.
4. Graphics adalah sebuah fasilitas yang dimiliki MATLAB untuk menampilkan vector dan matriks sebagai suatu grafik. Didalamnya melibatkan fungsi-fungsi level tinggi untuk visualisasi data dua dimensi dan data tiga dimensi, image processing, animation, dan presentation graphics. Ini juga melibatkan fungsi level rendah yang memungkinkan bagi kita untuk membiasakan diri untuk memunculkan grafik mulai dari bentuk yang sederhana sampai dengan tingkatan graphical user interfaces pada aplikasi MATLAB kita .
5. MATLAB Application Program Interface (API). Merupakan suatu library yang memungkinkan program yang telah kita tulis dalam bahasa C dan Fortran mampu berinteraksi dengan MATLAB. Ini melibatkan fasilitas untuk pemanggilan routines dari MATLAB (dynamic linking), pemanggilan MATLAB sebagai sebuah computational engine, dan untuk membaca dan menuliskan MAT-files.

## B. Fungsi dan Perintah Dasar Matlab

Sebagai aplikasi yang tergolong dalam bahasa pemrograman tingkat tinggi matlab memiliki berbagai macam fungsi dan perintah dasar idalamnya, berikut ini adalah fungsi dan perintah dasar yang terdapat didalam matlab.

**Tabel 2.1** Perintah Umum

Operator dan karakter Spesial	
+	operator tambahan.
-	operator pengurangan.
*	Operator perkalian skalar dan matriks.
.*	Operator perkalian array.
^	Operator eksponensial skalar dan matriks.
.^	Operator eksponensial array.
\	Operator divisi kiri.
/	Operator pembagian kanan.
.\	Array operator pembagian kiri.
./	Array operator pembagian kanan.
:	menghasilkan elemen spasi teratur dan mewakili seluruh baris atau kolom.
()	menyertakan argumen fungsi dan indeks array; mengesampingkan prioritas.
[]	elemen array lampiran.
.	Titik desimal.
...	operator kelanjutan garis.
„	memisahkan pernyataan dan elemen dalam satu baris
;	memisahkan kolom dan menekan tampilan.
%	menunjuk komentar dan menentukan pemformatan.
_	Tkita kutip dan operator transpos.
._	Operator transpos tak terkonjugasi.
=	Operator penugasan (penggantian).

Perintah Untuk Mengelola Sesi	
clc	Membersihkan jendela Perintah.
clear	Menghapus variabel dari memori.
exist	Memeriksa keberadaan file atau variabel.
global	Mendeklarasikan variabel menjadi global.
help	Mencari topik bantuan.
lookfor	Mencari entri bantuan untuk kata kunci.
quit	Menghentikan MATLAB.
who	Daftar variabel saat ini.
whos	Daftar variabel saat ini (tampilan panjang).
Variabel Khusus dan Konstanta	
ans	Jawaban terbaru.
eps	Akurasi presisi floating-point.
i,j	Satuan imajiner -1.
Inf	Ketakterbatasan.
NaN	Hasil numerik yang tidak ditentukan (bukan angka).
pi.	Nomor p.
ans	Jawaban terbaru.
Perintah Sistem dan File	
cd	Mengubah direktori saat ini.
date	Menampilkan tanggal saat ini.
delete	Menghapus file.
diary	Mengaktifkan/menonaktifkan perekaman file buku harian.
dir	Daftar semua file dalam direktori saat ini.
load	Memuat variabel ruang kerja dari file.
path	Menampilkan jalur pencarian.
pwd	Menampilkan direktori saat ini.
save	Menyimpan variabel ruang kerja dalam file.
type	Menampilkan isi file.
what	Daftar semua file MATLAB di direktori saat ini.
wklread	Membaca file spreadsheet .wk1.

**Tabel 2.2** Perintah Input/Output dan Pemformatan

<b>Perintah Input/Output</b>	
disp	Menampilkan isi array atau string.
fscanf	Membaca data yang diformat dari file.
format	Mengontrol format tampilan layar.
fprintf	Melakukan penulisan yang diformat ke layar atau file.
input	Menampilkan prompt dan menunggu input.
;	Menekan screen printing.
<b>Format Codes for fprintf and fscanf</b>	
%s	Format sebagai string.
%d	Format sebagai bilangan bulat.
%f	Format sebagai nilai floating point.
%e	Format sebagai nilai floating point dalam notasi ilmiah.
%g	Format dalam bentuk paling ringkas: %f atau %e.
\n	Sisipkan baris baru di string keluaran.
\t	Sisipkan tab di string keluaran.
<b>Format Tampilan Numerik</b>	
format short	format pendek Empat angka desimal (default).
format long	format panjang 16 digit desimal.
format short e	format pendek e Lima digit ditambah eksponen.
format long e	format panjang e 16 digit ditambah eksponen.
format bank	format bank Dua angka desimal.
format +	format + Positif, negatif, atau nol.
format rat	Pendekatan rasional.
format compact	Menekan beberapa umpan baris.
format loose	Setel ulang ke mode tampilan yang kurang ringkas

**Tabel 2.3** Perintah Vector, Matrix and Array

<b>Perintah Array</b>	
cat	Menggabungkan array.
find	Menemukan indeks elemen bukan nol.
length	Komputer jumlah elemen.
linspace	Membuat vektor dengan jarak teratur.
logspace	Membuat vektor dengan spasi logaritmik.
max	Mengembalikan elemen terbesar.
min	Mengembalikan elemen terkecil.
prod	Produk setiap kolom.
reshape	Ubah ukuran
size	Menghitung ukuran array.
sort	Urutkan setiap kolom.
sum	Jumlahkan setiap kolom.
<b>Matrik Spesial</b>	
eye	Membuat matriks identitas
ones	Membuat array yang satu
zeros	Membuat array nol
<b>Matrik Arithmatik</b>	
cross	Menghitung produk silang.
dot	Menghitung produk titik.
<b>Perintah Matriks untuk Menyelesaikan Persamaan Linier</b>	
det	Menghitung determinan array.
inv	Menghitung invers suatu matriks.
pinv	Menghitung pseudoinverse dari sebuah matriks.
rank	Menghitung rank suatu matriks.
rref	Menghitung bentuk eselon baris tereduksi.
<b>Fungsi Cell Array</b>	
cell	Membuat array sel.
celldisp	Menampilkan larik sel.
cellplot	Menampilkan representasi grafis dari array sel.
num2cell	Mengonversi array numerik ke array sel.
deal	Mencocokkan daftar input dan output.
iscell.	Mengidentifikasi array sel.

Fungsi Structur	
fieldnames	Mengembalikan nama bidang dalam array struktur.
Getfield	Mengembalikan konten bidang dari larik struktur.
isfield	Mengidentifikasi bidang array struktur.
isstruct	Mengidentifikasi array struktur.
rmfield	Menghapus bidang dari larik struktur.
setfield	Mengatur isi bidang.
struct	Membuat susunan struktur.

**Tabel 2.4** Perintah Plotting

Perintah Dasar Plotting xy	
axis	Menetapkan batas sumbu.
fplot	Merencanakan fungsi dengan cerdas.
grid	Menampilkan garis kisi.
plot	Menghasilkan plot xy.
print	Mencetak plot atau menyimpan plot ke file
title	Menempatkan teks di atas plot.
xlabel	Menambahkan label teks ke sumbu x.
ylabel	Menambahkan label teks ke sumbu y.
Perintah Peningkatan Plot	
axes	Membuat objek sumbu.
close	Menutup plot saat ini.
close all	Menutup semua plot.
figure	Membuka jendela gambar baru.
gtext	Mengaktifkan penempatan label dengan mouse.
hold	Membekukan plot saat ini.
Legend	Penempatan legenda dengan mouse.
refresh	Menggambar ulang jendela gambar saat ini.
set	Menentukan properti objek seperti sumbu.
subplot	Membuat plot di subwindows.
text	Tempatkan string pada gambar.

Perintah Khusus Plot	
bar	Membuat diagram batang.
loglog	Membuat plot log-log.
polar	Menciptakan plot kutub.
semilogx	Membuat plot semilog (absis logaritma).
semilogy	Membuat plot semilog (ordinat logaritmik).
stairs	Membuat pot tangga.
stem	Membuat plot batang.
Perintah Plotting 3-D	
contour	Membuat plot kontur.
mesh	Membuat plot permukaan mesh tiga dimensi.
meshc	Sama seperti mesh dengan plot kontur di bawahnya.
meshz	Sama seperti jala dengan garis vertikal di bawahnya.
plot3	Membuat plot tiga dimensi dari garis dan titik.
surf	Membuat plot permukaan mesh tiga dimensi yang diarsir.
surfc	Sama seperti surfing dengan kontur plot di bawahnya.
meshgrid	Membuat grid persegi panjang.
waterfall	Sama seperti mesh dengan garis mesh dalam satu arah.
zlabel	Menambahkan label teks ke sumbu z.
Fungsi Histogram	
bar	Membuat diagram batang.
hist	Menggabungkan data ke dalam tempat sampah dengan jarak yang sama.
histc.	Menggabungkan data ke dalam tempat sampah dengan jarak yang tidak sama.



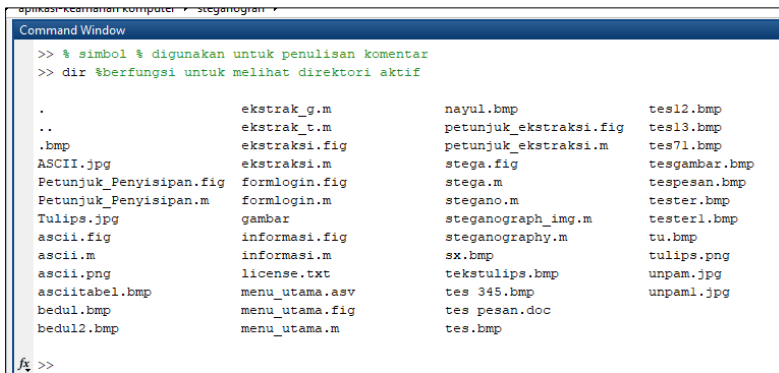
**Tabel 2.5 Programming**

<b>Operator Logika dan Relasional (Penghubung)</b>	
= =	Operator relasional: sama dengan.
~=	Operator relasional: tidak sama dengan.
<	Operator relasional: kurang dari.
<=	Operator relasional: kurang dari atau sama dengan.
>	Operator relasional: lebih besar dari.
>=	Operator relasional: lebih besar dari atau sama dengan.
&	Operator Logika: DAN.
	Operator Logika: ATAU.
~	Operator Logika: TIDAK.
xor	Operator Logika: EKSKLUSIF ATAU.
<b>Kontrol Flow Program</b>	
break	Mengakhiri eksekusi looping (perulangan).
case	Menyediakan jalur eksekusi alternatif dalam struktur switch.
else	Menggambarkan blok pernyataan alternatif.
elseif	Mengeksekusi pernyataan secara kondisional.
end	Mengakhiri pernyataan for, while, dan if.
error	Menampilkan pesan kesalahan.
for	Mengulangi pernyataan beberapa kali
if	Mengeksekusi pernyataan secara kondisional.
otherwise	Bagian default dari pernyataan switch.
return	Kembali ke fungsi pemanggilan.
switch	Mengarahkan eksekusi program dengan membandingkan titik dengan ekspresi kasus.
warning	Menampilkan pesan peringatan.
while	Mengulangi pernyataan dalam jumlah yang tidak terbatas.
<b>Fungsi Logika</b>	
any	Benar jika ada elemen yang bukan nol.
all	Benar jika semua elemen bukan nol.
find	Menemukan indeks elemen bukan nol.

finite	Benar jika elemennya berhingga.
isnan	Benar jika elemen tidak terdefinisi.
isinf	Benar jika elemen tidak terbatas.
isempty	Benar jika matriks kosong.
isreal.	Benar jika semua elemen nyata.
<b>M-Files</b>	
eval	Menggambarkan string yang berisi ekspresi Matlab.
feval	Evaluasi fungsi.
function	Membuat file-M fungsi yang ditentukan pengguna.
global	Tentukan variabel global.
nargin	Jumlah argumen masukan fungsi.
nargout	Jumlah argumen keluaran fungsi.
script	Script M-files
<b>Pengaturan Waktu</b>	
cputime.	waktu CPU dalam hitungan detik.
clock	Tanggal dan waktu saat ini sebagai vektor tanggal.
tic, toc.	Mulai, hentikan penghitung waktu stopwatch.

### C. Command Window

Command window merupakan fasilitas dari matlab berfungsi untuk mengakses atau menjalankan perintah-perintah(scripts) dan fungsi-fungsi yang ada didalam MATLAB. Untuk mengakses perintah-perintah dapat dilakukan secara langsung dengan cara mengetikkan pada command window MATLAB.



**Gambar 2.1** Command Window

Didalam MATLAB Perintah dapat berupa fungsi-fungsi pengaturan file (seperti perintah DOS/UNIX) maupun fungsi-fungsi bawaan atau toolbox MATLAB sendiri. Berikut ini beberapa contoh penggunaan perintah dan fungsi pengaturan file dalam MATLAB:

Perintah yang di eksekusi secara langsung dan outputnya ditampilkan pada command window (diakhiri tanpa ;)

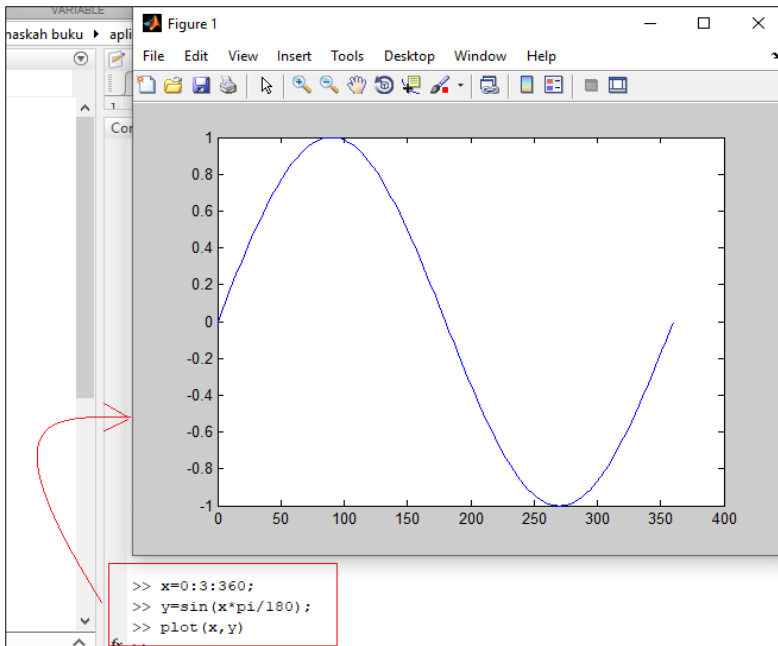
```
>> ls
Petunjuk_Penyisipan.fig informasi.m tes.bmp
Petunjuk_Penyisipan.m license.txt tes12.bmp
Tulips.jpg menu_utama.asv tes13.bmp
>> cd gambar
>> whos
Name Size Bytes Class Attributes
A 3x3 72 double
B 3x3 72 double
C 3x3 72 double
a 1x9 72 double
ans 1x1 16 double complex
b 3x1 24 double
p 1x4 32 double
q 1x7 56 double
r 1x10 80 double
x 3x1 24 double
```

```
>> ans  
ans =  
0.0000 + 1.0000i
```

Perintah yang di eksekusi secara langsung dan outputnya tidak ditampilkan pada command window( diakhiri dengan ;)

```
>> x=0:3:360;  
>> y=sin(x*pi/180);  
>> plot(x,y)
```

Dengan menjalankan perintah diatas maka ouput yang dihasilkan, seperti gambar dibawah ini.



**Gambar 2.2** Tampilan Grafik Sinus

Dalam bahasa pemrograman untuk menginisialisasi suatu nilai awal perlu dilakukan sebuah deklarasi variabel. Dalam pembuatan atau pendeklarasian suatu variabel, bisa menggunakan satu karakter, satu kata bahkan dua kata atau

lebih (menggunakan garis bawah sebagai pemisah kata). Gambar dibawah ini adalah contoh pembuatan variabel didalam matlab.

```
Command Window

>> %contoh pembuatan variabel dengan 1 karakter
>> x=1;
>> y=3;
>> x*y

ans =

    3

>> %contoh variabel dengan 1 kata
>> satu=2;
>> kata=6;
>> satu+kata

ans =

    8

>> %contoh variabel dengan lebih dari satu kata
>> bilangan_genap=6;
>> bilangan_ganjil=5;
>> bilangan_ganjil/bilangan_genap

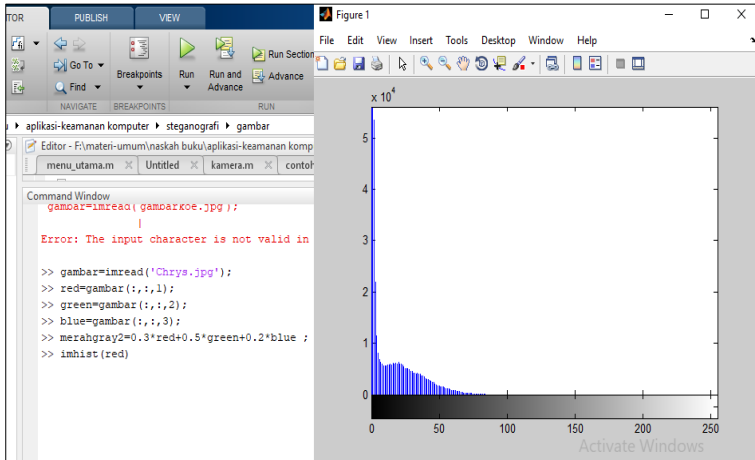
ans =

    0.8333
```

Dibawah ini adalah contoh script atau perintah yang berfungsi untuk membaca gambar dan menampilkan histogram gambar yang di jalankan didalam command window.

```
>> gambar=imread('Chrys.jpg');
>> red=gambar(:, :,1);
>> green=gambar(:, :,2);
>> blue=gambar(:, :,3);
>> merahgray2=0.3*red+0.5*green+0.2*blue ;
>> imhist(red)
```

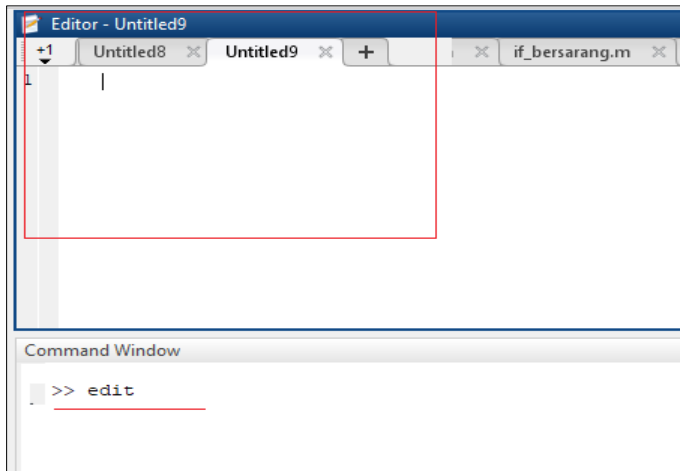
Output yang ditampilkan dari baris perintah diatas akan menampilkan histogram gambar n dari warna merah.



**Gambar 2.3** Contoh histogram

#### D. M-Files

Matlab memberikan kemudahan di dalam penulisan script atau perintah-perintah. Script atau perintah tersebut dapat dituliskan di dalam editor tertentu dan disimpan ekstensi .m. Dengan memberikan nama file dengan ekstensi tersebut, maka akan dapat deksekusi di dalam lingkungan Matlab. Juga, dengan ekstensi tersebut file yang kita buat bernama M-File. M-File dapat berupa skrip maupun fungsi.



**Gambar 2.4** Tampilan Editor Matlab

M\_File dalam bentuk skrip dapat dieksekusi secara mudah dengan menuliskan nama skrip tersebut di command window, atau cukup menekan tombol F5 dengan catatan skrip sudah diberikan nama. M-File dalam bentuk fungsi terdiri atas argumen yang harus diberikan input sebelum dieksekusi dan menghasilkan output. Kita dapat menampilkan jendela editor untuk membuat M-file dengan menjalankan perintah edit didalam command window.

### #Latihan 1-M.files- FUNGSI MATEMATIKA

Pada bagian kita akan membuat fungsi matematika sederhana didalam editor matlab, silahkan klik file/new pada tab editor, untuk membuat file baru, kemudian masukan source code fungsi dibawah ini.

```
function contoh_MTK()
a=input('masukan nilai a =');
b=input('masukan nilai b =');
c=input('masukan nilai c =');
disp([a,b,c])
d=b^2-4*a*c;
x1=(-b+sqrt(d))/(2*a)
```


```

pause(5)
x2=(-b-sqrt(d))/(2*a)
pause(3)
x=[1:1:10]
y=a.*x.^2+b.*x+c;
plot(x,y)

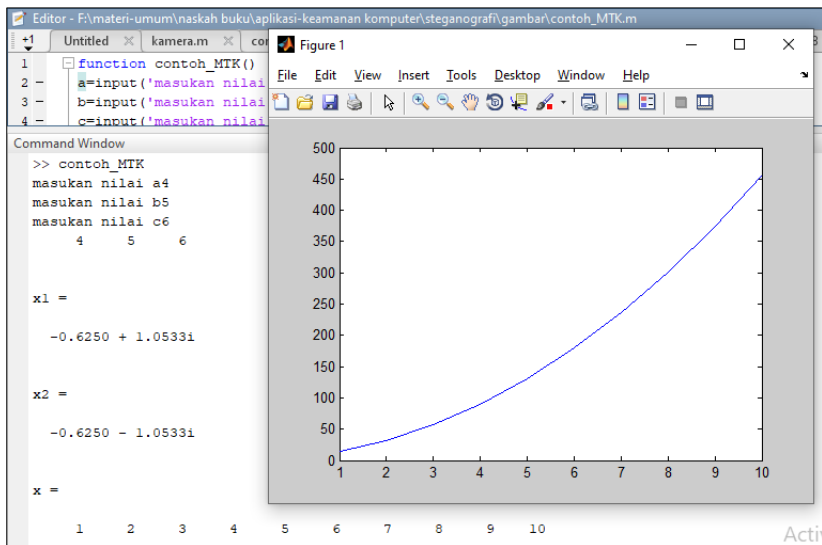
```

**Penjelasan code:**

- ✓ **function...:pemanggilan nama sebuah fungsi**
- ✓ **a= : variabel yang digunakan untuk menampung sebuah nilai**
- ✓ **input: fungsi atau perintah yang digunakan memasukan sebuah nilai masukan dari pengguna pada command window**
- ✓ **disp : fungsi atau perintah untuk menampilkan sebuah nilai atau variabel pada command window**
- ✓ **sqrt: fungsi matematika dasar untuk mencari akar kuadrat**
- ✓ **pause : fungsi untuk berhenti sejenak**
- ✓ **plot : fungsi untuk menempatkan suatu grafik ataupun gambar**

Simpan dengan nama contoh\_MTK.m lalu tekan f5 atau bisa juga dengan mengklik  icon pada tab editor, untuk menjalankan aplikasi, maka output yang ditampilkan akan seperti gambar dibawah ini.





**Gambar 2.5** Tampilan Hasil Latihan 1

Saat aplikasi di jalankan, maka pengguna akan di minta memasukan nilai setiap variabel pada command windows, untuk kemudian dilakukan perhitungan matematis dan hasilnya akan di representasikan kedalam bentuk kurva atau grafik.

### #Latihan 2-M.files- MENGHITUNG KELILING LINGKARAN

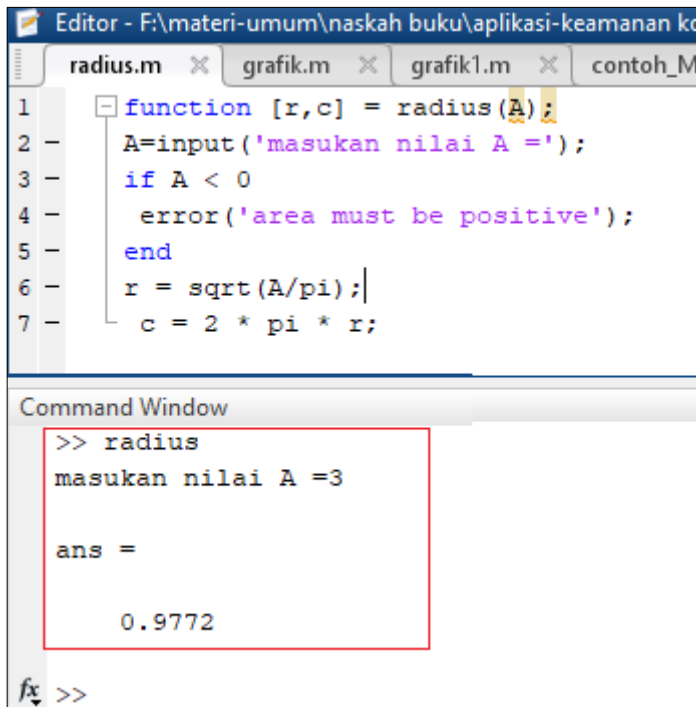
Pada latihan membuat M.File kedua ini, kita akan mencoba mengaplikasikan bentuk perhitungan keliling lingkaran, dan juga menggunakan sebuah fungsi kondisional If-End. Buatlah sebuah file baru, kemudian masukan fungsi dibawah ini.

```

function [r,c] = radius(A);
A=input('masukan nilai A =');
if A < 0
    error('area must be positive');
end
r = sqrt(A/pi);
c = 2 * pi * r;

```

Simpan dengan nama radius.m, kemudian jalankan atau running fungsi tersebut. Output yang dihasilkan akan seperti gambar dibawah ini.



The screenshot shows the MATLAB Editor window with the file 'radius.m' open. The code in the editor is as follows:

```
1 function [r,c] = radius(A);  
2 A=input('masukan nilai A =');  
3 if A < 0  
4     error('area must be positive');  
5 end  
6 r = sqrt(A/pi);  
7 c = 2 * pi * r;
```

Below the editor is the Command Window, which shows the execution of the function:

```
>> radius  
masukan nilai A =3  
  
ans =  
  
    0.9772
```

A red rectangle highlights the input and output in the Command Window.

**Gambar 2.6** Tampilan Hasil Latihan 2

Ketika fungsi radius.m dijalankan maka pengguna akan di minta memasukan sebuah nilai didalam command window, dimana nilai tersebut harus berupa bilangan positif.

### #Latihan 3-M.files- OPERATOR ARITMATIKA DASAR

Di latihan ketiga ini kita akan membuat sebuah fungsi penggunaan operator aritmatika dasar untuk menghitung sebuah nilai, buat file baru, lalu kemudian masukan source code di bawah ini.

```
function nilaiMhs()  
clear all;
```

```

clc;
disp('-----');
disp('aplikasi penghitung nilai ');
disp('Matlab Programming');
disp('Created By Dzaya 2022');
disp('-----');
tugas=input('nilai tugas =');
uts=input('nilai uts =');
uas=input('nilai uas =');
na=(tugas*20/100)+(uts*30/100)+(uas*50/100);
disp(['nilai akhir = ' num2str(na)]);
if na>80 && na<=100
    disp('grade kita =A');
elseif na<=80 && na>70
    disp('grade kita =B');
elseif na<=70 && na>60
    disp('grade kita =C');
elseif na<=60 && na>50
    disp('grade kita =D');
else
    disp('grade kita =E');
end

```

**penjelasan code:**

**clear all;** menghapus semua variabel dari memori

**clc;** perintah untuk membersihkan layar pada command window

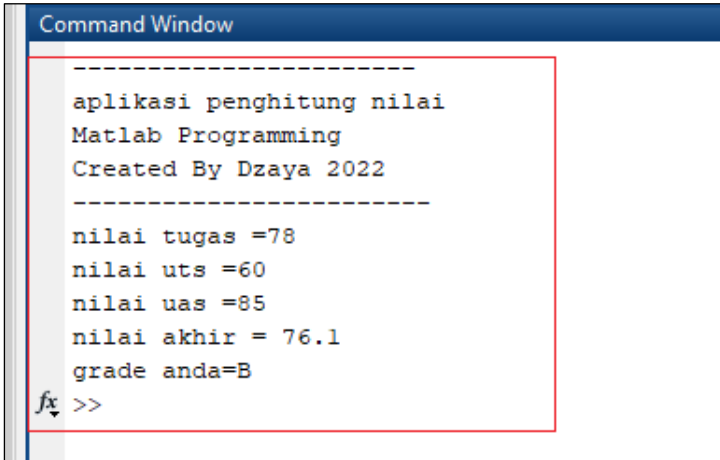
**>** :operator lebih besar dari

**<** :operator lebih kecil dari

**&&** : operator penghubung

**if, elseif, else** : control flow yang bersifat kondisional, dimana sebuah aplikasi atau program akan mengeksekusi sekelompok pernyataan ketika ekspresi benar.

Output dari aplikasi akan menjalankan operasi aritmatika dasar berdasarkan kepada nilai-nilai yang telah di masukan.



```
-----
aplikasi penghitung nilai
Matlab Programming
Created By Dzaya 2022
-----
nilai tugas =78
nilai uts =60
nilai uas =85
nilai akhir = 76.1
grade anda=B
fx >>
```

**Gambar 2.7** Tampilan Hasil Latihan 3

#### **#Latihan 4-M.files- FUNGSI PENGAMBARAN GRAFIK**

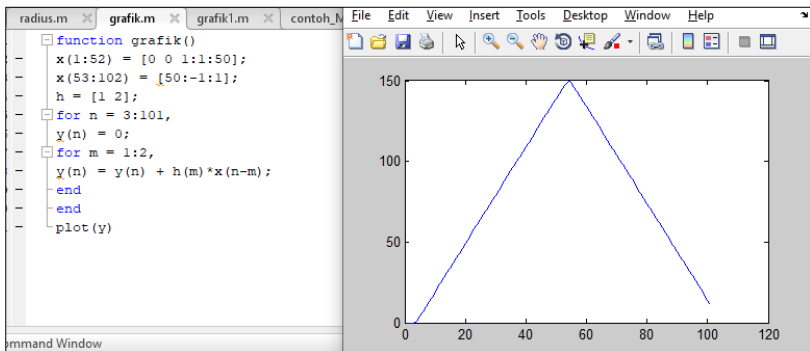
Untuk menggambarkan grafik dua dimensi dapat menggunakan perintah plot, perintah ini berfungsi untuk menggambarkan data dalam array pada sumbu yang bersesuaian dan menghubungkan titik-titik tersebut dengan garis lurus. Pada latihan kali ini kita akan mencoba melakukan visualisasi data dengan menggunakan grafik sederhana.

```

function grafik()
    x(1:52) = [0 0 1:1:50];
    x(53:102) = [50:-1:1];
    h = [1 2];
    for n = 3:101,
        y(n) = 0;
    for m = 1:2,
        y(n) = y(n) + h(m)*x(n-m);
    end
    end
    plot(y)

```

Masukan source code diatas, lalu simpan dengan nama grafik.m. jalankan file tersebut dan perhatikan output yang dihasilkan



**Gambar 2.8** Tampilan Hasil Latihan 4

### #Latihan 5-M.files- PENGOLAHAN GAMBAR

Image atau gambar adalah representasi spasial dari suatu objek yang sebenarnya dalam bidang dua dimensi yang biasanya ditulis dalam koordinat kartesian x-y, dan setiap koordinat merepresentasikan satu sinyal terkecil dari objek yang biasanya koordinat terkecil ini disebut sebagai piksel. Pada matlab memiliki fungsi-fungsi untuk melakukan pengolahan gambar, pada latihan kali ini kita akan membuat sebuah aplikasi

untuk membaca gambar dan menampilkan histogram dari gambar tersebut.

```
function baca_gambar()
gambar=imread('Chrys.jpg');
red=gambar(:,:,1);
green=gambar(:,:,2);
blue=gambar(:,:,3);
merahgray2=0.3*red+0.5*green+0.2*blue ;
figure, imhist(red)
figure, imhist(blue)
figure, imshow(gambar);
```

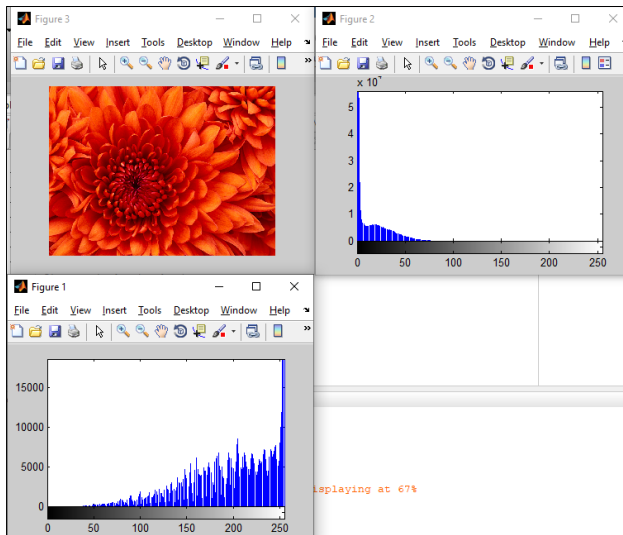
**penjelasan code:**

**imread** : fungsi untuk membaca gambar

**imshow** : fungsi untuk menampilkan gambar

**imhist** : berfungsi untuk menampilkan histogram dari gambar

Masukan sourcecode diatas kemudian simpan dengan nama `baca_gambar.m`, lalu running aplikasi tersebut, output dari aplikasi tersebut akan menampilkan histogram dari gambar berdasarkan warna, seperti gambar dibawah ini.



**Gambar 2.9** Tampilan Hasil Latihan 5

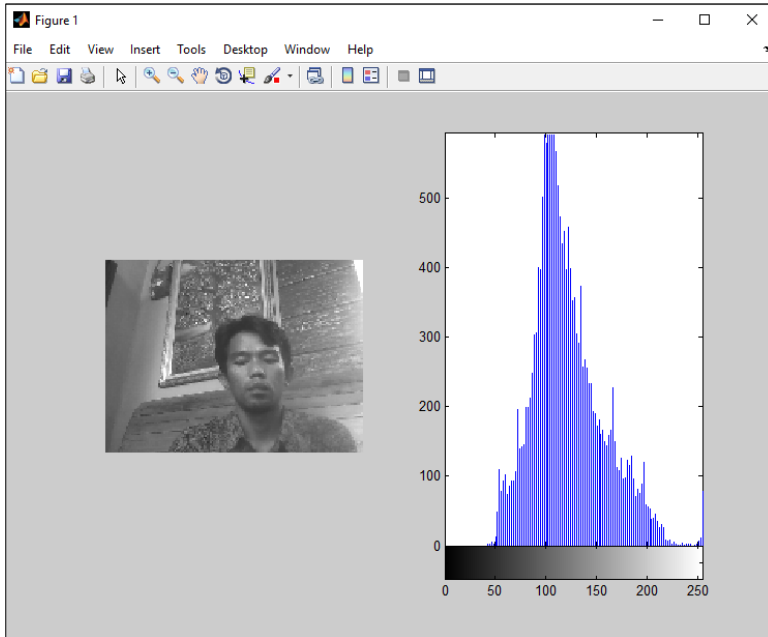
## #Latihan 6-M.files- FUNGSI MENGAKSES KAMERA

Dalam bidang pengolahan citra digital, histogram sangat penting untuk proses analisis data. Selain histogram dari file gambar, histogram live atau histogram yang merupakan interpretasi dari objek berbentuk video.

Pada bagian kita akan mengakses webcam pada komputer kita, dan kemudian membuat live histogram dari video yang ditangkap oleh kamera webcam kita. Buatlah file baru kemudian masukan sourcecode dibawah ini.

```
function kamera()
    utilpath = fullfile(matlabroot, 'toolbox', 'imaq',
        'imaqdemos', 'helper');
    addpath(utilpath);
    vidobj = videoinput('winvideo');
    set(vidobj, 'ReturnedColorSpace', 'grayscale')
    vidRes = get(vidobj, 'VideoResolution');
    f = figure('Visible', 'off');
    imageRes = fliplr(vidRes);
    subplot(1,2,1);
    hImage = imshow(zeros(imageRes));
    axis image;
    setappdata(hImage, 'UpdatePreviewWindowFcn', @update_livehistogram_display);
    preview(vidobj, hImage);
    pause(15);
```

Simpanlah file tersebut dengan nama kamera.m, output dari aplikasi tersebut akan mengakses webcam kita , kemudian akan menangkap objek video untuk kemudian menampilkan objek video tersebut dan membuat live histogram.



**Gambar 2.10** Tampilan Hasil Latihan 7

## E. Control Flow

Control flow merupakan sebuah pengendali atau pengontrol aliran proses program. Dalam membuat sebuah aplikasi atau program pasti akan memiliki tingkat kompleksitas tersendiri, dimana didalamnya akan terdapat satu atau banyak kondisi ataupun pernyataan. Oleh karena itu control flow memiliki peranan yang sangat penting dalam pembuatan suatu aplikasi atau program. Didalam matlab terdapat beberapa jenis control flow, yaitu:

### 1. Control Flow If, Elseif, Else

if, elseif, else adalah control flow yang bersifat kondisional, dimana sebuah aplikasi atau program akan mengeksekusi sekelompok pernyataan ketika ekspresi benar. Bentuk dasar penulisan sintaks dari control flow ini adalah sebagai berikut:



```
if expression
    statements
elseif expression
    statements
else
    statements
end
```

Sintaks elseif dan else adalah pilihan atau opsional, dan pernyataan ini dijalankan hanya ketika ekspresi sebelumnya di blok if salah. Sebuah blok if dapat menyertakan beberapa pernyataan elseif. Sebuah ekspresi yang dievaluasi akan bernilai benar ketika hasilnya tidak kosong dan berisi semua elemen bukan nol (logis atau numerik nyata), jika tidak, ekspresinya akan bernilai salah.

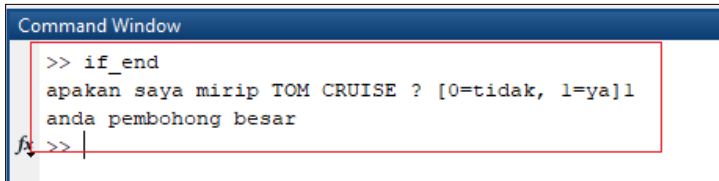
Sebuah ekspresi dapat menyertakan operator relasional (seperti < atau ==) dan operator Logika (seperti &&, ||, atau ~). Dalam ekspresi kondisi pernyataan if atau while, operator Logika & dan | berfungsi sebagai operator penghubung singkat, fungsi ini sama dengan && dan ||. Tetapi disarankan untuk praktik yang baik adalah menggunakan && dan || bukannya & dan | dalam ekspresi, karena dirasa lebih konsisten dalam ekspresi dan pernyataan kondisi if dan while.

#### #LATIHAN 1. STATEMEN IF-END

Silahkan kita ketikkan fungsi pernyataan if-end dibawah ini pada jendela editor matlab.

```
function if_end()
zonk='kita pembohong besar';
tanya= input('apakah saya mirip TOM CRUISE ? [0=tidak, 1=ya]');
if (tanya==1)
    disp(zonk);
end
```

Simpan file tersebut dengan nama if-end.m. Kemudian jalankan aplikasi tersebut, maka output yang akan dihasilkan akan seperti gambar dibawah ini

A screenshot of a MATLAB Command Window. The title bar says "Command Window". The command prompt shows the execution of an if-end statement: 

```
>> if_end  
apakah saya mirip TOM CRUISE ? [0=tidak, 1=ya]1  
anda pembohong besar  
fx >> |
```

 A red rectangle highlights the command and its output.

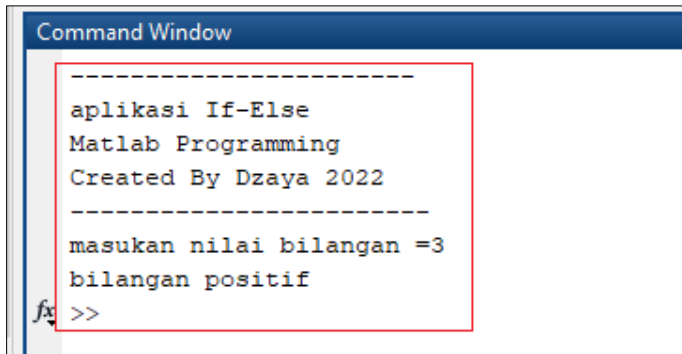
**Gambar 2.11** Tampilan Hasil Fungsi If

## #LATIHAN 2. STATEMEN IF ELSE-END

Silahkan kita ketikkan fungsi pernyataan if-else-end dibawah ini pada jendela editor matlab

```
function contoh_if()  
clear all;  
clc;  
disp('-----');  
disp('aplikasi If-Else ');  
disp('Matlab Programming');  
disp('Created By Dzaya 2022');  
disp('-----');  
a=input('masukan nilai bilangan =');  
if(abs(a)==a)  
    disp('bilangan positif');  
else  
    disp('bilangan negatif');  
end;
```

Simpan file tersebut dengan nama contoh\_if.m. Kemudian jalankan aplikasi tersebut, maka output yang akan dihasilkan akan seperti gambar dibawah ini

A screenshot of the MATLAB Command Window. The window has a blue title bar that says "Command Window". Inside, there is a red rectangular box highlighting the output of a function. The output consists of several lines of text: a dashed line, "aplikasi If-Else", "Matlab Programming", "Created By Dzaya 2022", another dashed line, "masukan nilai bilangan =3", "bilangan positif", and a prompt ">>".

```
-----  
aplikasi If-Else  
Matlab Programming  
Created By Dzaya 2022  
-----  
masukan nilai bilangan =3  
bilangan positif  
fx >>
```

**Gambar 2.12** Tampilan Hasil fungif-else 1

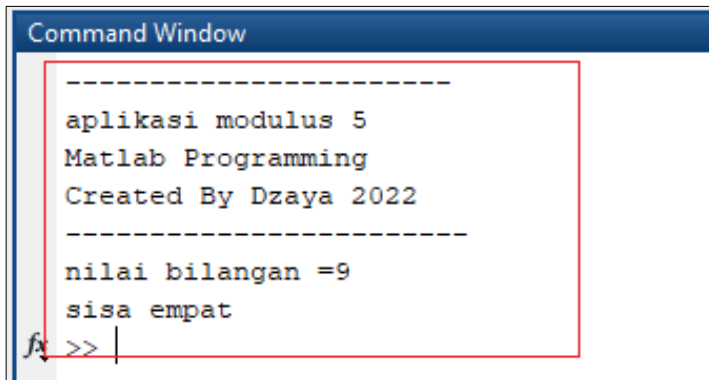
### #LATIHAN 3. STATEMEN IF ELSE-END

Silahkan kita ketikkan fungsi pernyataan if-elseif-else-end dibawah ini pada jendela editor matlab.

```
function if_bersarang()  
clear all;  
clc;  
disp('-----');  
disp('aplikasi modulus 5 ');  
disp('Matlab Programming');  
disp('Created By Dzaya 2022');  
disp('-----');  
m = input('nilai bilangan =');  
n = mod(m,5);  
if(n==0)%konstanta 0  
    disp('tak bersisa');  
elseif (n==1)%konstanta 1  
    disp('sisa satu');  
elseif (n==2)%konstanta 2  
    disp('sisa dua');  
elseif (n==3)%konstanta 3  
    disp('sisa tiga');  
elseif (n==4)%konstanta 4  
    disp('sisa empat');  
else
```

```
fprintf('m bilangan pecahan,sisa %f\n',n);  
end;
```

Simpan file tersebut dengan nama if\_bersarang.m. Kemudian jalankan aplikasi tersebut, maka output yang akan dihasilkan akan seperti gambar dibawah ini.



```
-----  
aplikasi modulus 5  
Matlab Programming  
Created By Dzaya 2022  
-----  
nilai bilangan =9  
sisanya empat  
fx >> |
```

Gambar 2.13 Tampilan Hasil fungsi if-else 2

## 2. Control Flow For

Perulangan adalah jenis control flow yang berguna untuk mengefisienkan penulisan script program, khususnya untuk program-program yang membutuhkan proses berulang-ulang. Perulangan ini sering juga disebut sebagai iterasi atau looping. Control flow ini akan mengeksekusi dan menjalankan sebuah pernyataan yang nilainya ditentukan beberapa kali, adapun bentuk dasar dari sintaks for pada matlab

```
for index = values  
    program statements  
:  
end
```

Fungsi perulangan ini akan mengeksekusi atau menjalankan satu atau lebih pernyataan dalam satu lingkaran perulangan. Pada matlab terdapat beberapa bentuk nilai yang biasa digunakan dalam perulangan (looping), yaitu:

a. `initval:endval`

Fungsi ini akan menambah indeks sebuah variabel dari `initval` ke `endval` sebesar 1, dan mengulangi eksekusi pernyataan program hingga indeks lebih besar dari `endval`.

b. `Initval:step:endval`

Fungsi ini akan menambah indeks dengan langkah nilai pada setiap iterasi, atau mengurangi ketika langkah negatif.

c. `valArray`

Fungsi ini akan membuat indeks vektor kolom dari kolom berikutnya dari array `valArray` pada setiap iterasi. Misalnya, pada iterasi pertama, `index = valArray(:,1)`. Loop dijalankan maksimal `n` kali, di mana `n` adalah jumlah kolom `valArray`, diberikan oleh `numel(valArray, 1, :)`. Input `valArray` dapat berupa tipe data MATLAB apa pun, termasuk string, array sel, atau struct.

#### #LATIHAN 4. PERULANGAN FOR-END

Silahkan kita ketikkan fungsi pernyataan for-end dibawah ini pada jendela editor matlab

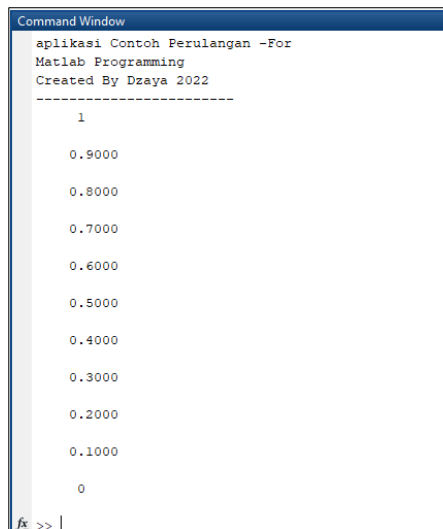
```
function contoh_for1()
clear all;
clc;
disp('-----');
disp('aplikasi Contoh Perulangan -For ');
disp('Matlab Programming');
```

```

disp('Created By Dzaya 2022');
disp('-----');
for s = 1.0: -0.1: 0.0
    disp(s)
end

```

Simpan file tersebut dengan nama contoh\_for1.m. Kemudian jalankan aplikasi tersebut, maka output yang akan dihasilkan akan seperti gambar dibawah ini.



```

Command Window
aplikasi Contoh Perulangan -For
Matlab Programming
Created By Dzaya 2022
-----
1
0.9000
0.8000
0.7000
0.6000
0.5000
0.4000
0.3000
0.2000
0.1000
0
fx >> |

```

**Gambar 2.14** Tampilan Latihan For

## #LATIHAN 5. PERULANGAN FOR-FOR-END

Silahkan kita ketikkan fungsi pernyataan for-end dibawah ini pada jendela editor matlab

```

function contoh_for()
clear all;
clc;
disp('-----');
disp('aplikasi Contoh Perulangan -For ');
disp('Matlab Programming');
disp('Created By Dzaya 2022');

```

```

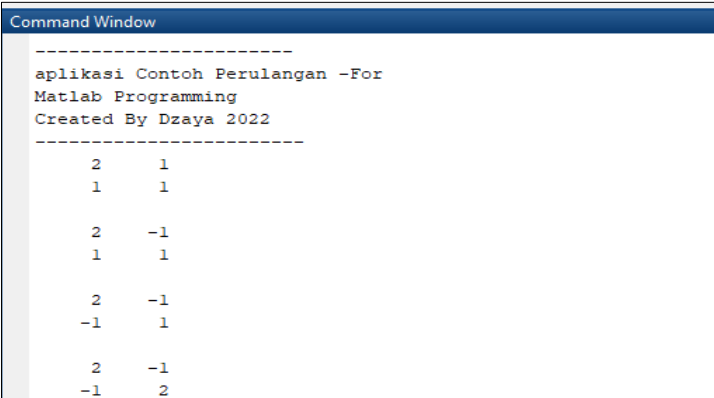
disp('-----');
% mengalokasikan nilai matrik
nrows = 2;
ncols = 2;
myData = ones(nrows, ncols);

% perulangan nilai matrik
for r = 1:nrows
    for c = 1:ncols

        if r == c
            myData(r,c) = 2;
        elseif abs(r - c) == 1
            myData(r,c) = -1;
        else
            myData(r,c) = 0;
        end
    end
end
disp(myData);
end
end

```

Simpan file tersebut dengan nama contoh\_for.m. Kemudian jalankan aplikasi tersebut, maka output yang akan dihasilkan akan seperti gambar dibawah ini.



```

Command Window

-----
aplikasi Contoh Perulangan -For
Matlab Programming
Created By Dzaya 2022
-----

    2     1
    1     1

    2    -1
    1     1

    2    -1
   -1     1

    2    -1
   -1     2

```

**Gambar 2.15** Tampilan Latihan For-End

### 3. Control Flow Parfor

Control flow ini berfungsi untuk melakukan proses paralel untuk fungsi perulangan (looping). Adapun bentuk dasar dari penulisan sintaks dasarnya adalah:

```
parfor loopvar = initval:endval; statements; end  
parfor (loopvar = initval:endval, M); statements; end
```

**Parfor loopvar = initval:endval; statements; end** : Fungsi ini digunakan untuk mengeksekusi serangkaian pernyataan dimana nilai dari loopvar antara initval dan endval, inklusif, yang menentukan vektor peningkatan nilai integer. Loop berjalan secara paralel saat Kita memiliki Parallel Computing Toolbox™ atau saat Kita membuat fungsi MEX dengan MATLAB Coder™.

**Parfor (loopvar = initval:endval, M); statements; end**

digunakan untuk menjalankan atau mengeksekusi pernyataan dalam satu lingkaran menggunakan maksimum M, di mana M adalah bilangan bulat non-negatif.

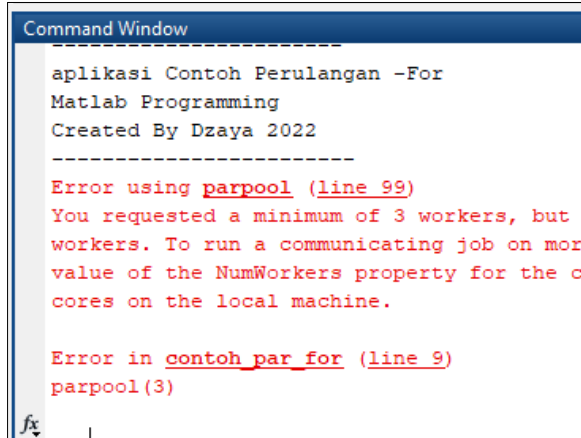
#### #LATIHAN 6. PERULANGAN PARFOR

Silahkan kita ketikkan fungsi pernyataan par-for dibawah ini pada jendela editor matlab

```
function contoh_par_for()
clear all;
clc;
disp('-----');
disp('aplikasi Contoh Perulangan -For ');
disp('Matlab Programming');
disp('Created By Dzaya 2022');
disp('-----');
parpool(3)
parfor i=1:3, c(:,i) = eig(rand(1000));
end
```



Simpan file tersebut dengan nama `contoh_par_for.m`. Kemudian jalankan aplikasi tersebut, maka output yang akan dihasilkan akan terjadi error, hal ini karena untuk melakukan perulangan paralel minimal kita harus menggunakan 2 mesin untuk bekerja.



**Gambar 2.16** Tampilan Latihan ParFor

#### 4. Control Flow Switch, Case, Otherwise

Control flow ini dalam beberapa kasus dijalankan atau di eksekusi berdasarkan dari ekspresi. Sebuah fungsi blok switch secara kondisional mengeksekusi satu set pernyataan dari beberapa pilihan, dimana setiap pilihan dianggap sebagai kasus. Adapun bentuk dasar penulisan sintaks switch case ini adalah sebagai berikut.

```
switch switch_expression
case case_expression
    statements
case case_expression
    statements
...
otherwise
    statements
end
```

Ekspresi didalam Switch (Switch\_expression) yang dievaluasi adalah skalar atau string, atau array sel skalar atau string. Blok switch menguji setiap kasus sampai salah satu kasus benar. Suatu kasus atau pilihan dikatakan benar ketika:

- a. Untuk angka, eq(case\_expression,switch\_expression).
- b. Untuk string, strcmp(case\_expression,switch\_expression).
- c. Untuk objek yang mendukung fungsi eq(case\_expression,switch\_expression).
- d. Untuk case\_expression array sel, setidaknya salah satu elemen array sel cocok dengan switch\_expression, seperti yang didefinisikan di atas untuk angka, string, dan objek.

Ketika sebuah kasus benar, aplikasi akan mengeksekusi pernyataan yang sesuai, dan kemudian keluar dari blok switch. Pernyataan otherwise hanya akan dijalankan atau dieksekusi jika tidak ada kasus atau pilihan yang benar.

#### #LATIHAN 7. SWITCH, CASE, OTHERWISE

Silahkan kita ketikkan fungsi pernyataan for-end dibawah ini pada jendela editor matlab

```
function switch_case()
clear all;
clc;
disp('-----');
disp('aplikasi Contoh Switch-Case ');
disp('Matlab Programming');
disp('Created By Dzaya 2022');
disp('-----');
mynumber = input('Masukan Angka (-1;0;1)=');

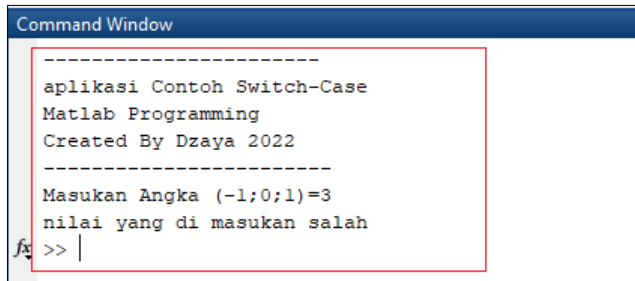
switch mynumber
    case -1
        disp('negative satu');
    case 0
        disp('nol/kosong');
    case 1
```

```

        disp('positive satu');
    otherwise
        disp('nilai yang di masukan salah');
end

```

Simpan file tersebut dengan nama contoh\_for.m. Kemudian jalankan aplikasi tersebut, maka output yang akan dihasilkan akan seperti gambar dibawah ini.



```

Command Window

-----
aplikasi Contoh Switch-Case
Matlab Programming
Created By Dzaya 2022
-----
Masukan Angka (-1;0;1)=3
nilai yang di masukan salah
fx >> |

```

**Gambar 2.17** Tampilan Latihan Switch-Case

## #LATIHAN 8. Swicth-Case

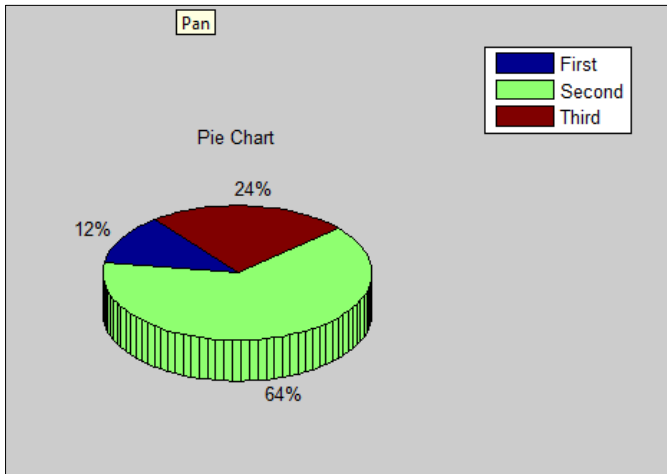
Berikut adalah contoh bentuk penggunaan fungsi switch case kedua pada file switch case.m.

```

function switch_case1()
x = [12, 64, 24];
plottype = 'pie3';

switch plottype
    case 'bar'
        bar(x)
        title('Bar Graph')
    case {'pie','pie3'}
        pie3(x)
        title('Pie Chart')
        legend('First','Second','Third')
    otherwise
        warning('Unexpected plot type. No plot created.');
```

Output dari aplikasi diatas ketika dijalankan akan menampilkan grafik pie berdasarkan data yang telah di inputkan.



**Gambar 2.18** Tampilan Latihan Switch-Case-Otherwise

## 5. Control Flow Try, Catch

Control flow try, catch ini memungkinkan kita untuk mengganti perilaku kesalahan default untuk serangkaian pernyataan program. Jika ada pernyataan dalam blok try yang menghasilkan kesalahan, kontrol program langsung menuju ke blok catch, yang berisi pernyataan penanganan kesalahan Kita . Berikut ini bentuk dasar penulisan sintaks try, catch.

```
Try
    Statements
catch exceptions
    statements
end
```

Exception adalah input objek MException opsional yang mengarah ke blok catch yang memungkinkan Kita mengidentifikasi kesalahan. Baik blok try dan catch dapat berisi pernyataan try/catch bersarang.

## #LATIHAN 9. TRY, CATCH

Berikut adalah contoh bentuk penggunaan fungsi try, catch

```
function contoh_try_catch()
A = rand(3);
B = ones(5);

try
    C = [A; B];
catch err

    % memberikan informasi tentang kesalahan
    if
        (strcmp(err.identifier, 'MATLAB:catenate:dimensionMismatch'))

        msg = ['Dimension mismatch occurred: First argument
        has ', ...
            num2str(size(A,2)), ' columns while second has ', ...
            num2str(size(B,2)), ' columns.'];
        error('MATLAB:myCode:dimensions', msg);

    % menampilkan apakah ada yang error
    else
        rethrow(err);
```

Penjelasan code: code diatas berfungsi untuk memberikan informasi selengkapny tentang kesalahan ketidakcocokan dimensi

## 6. Control Flow While

Control flow while akan menjalankan atau mengeksekusi satu atau lebih pernyataan program dalam satu lingkaran selama ekspresi tetap benar. Ekspresi yang dievaluasi benar ketika hasilnya tidak kosong dan berisi semua elemen bukan nol (logis atau numerik nyata). Jika tidak, ekspresinya bernilai salah.

**while expression  
statements  
end**

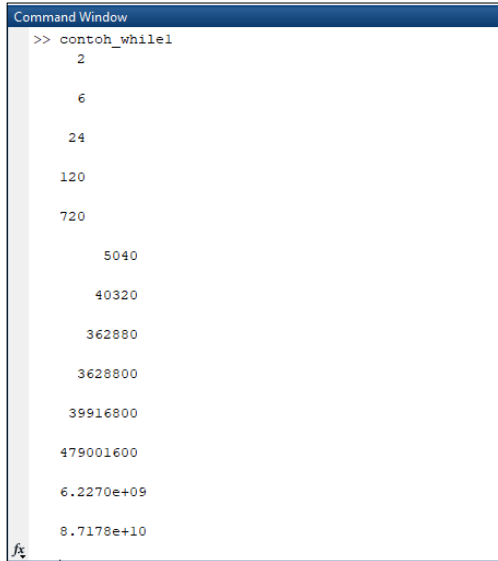
Sebuah ekspresi dapat menyertakan operator relasional (seperti < atau ==) dan operator Logika (seperti &&, ||, atau ~). Dalam ekspresi kondisi pernyataan if atau while, operator Logika & dan | berfungsi sebagai operator penghubung singkat, fungsi ini sama dengan && dan ||. Tetapi disarankan untuk praktik yang baik adalah menggunakan && dan || bukannya & dan | dalam ekspresi, karena dirasa lebih konsisten dalam ekspresi dan pernyataan kondisi if dan while.

#### #LATIHAN 10. WHILE

Silahkan kita ketikkan fungsi pernyataan while dibawah ini pada jendela editor matlab

```
function contoh_while1()
n = 1;
nFactorial = 1;
while nFactorial < 1e10
    n = n + 1;
    nFactorial = nFactorial * n;
disp(nFactorial);
end
```

Hasil dari aplikasi diatas adalah seperti di bawah ini

A screenshot of the MATLAB Command Window. The title bar says "Command Window". The prompt is ">>". The first line of code is "contoh\_while1". Below it, the output of the loop is displayed as a list of numbers: 2, 6, 24, 120, 720, 5040, 40320, 362880, 3628800, 39916800, 479001600, 6.2270e+09, and 8.7178e+10. The window has a standard MATLAB interface with a blue title bar and a light gray background.

```
Command Window
>> contoh_while1
2
6
24
120
720
5040
40320
362880
3628800
39916800
479001600
6.2270e+09
8.7178e+10
```

**Gambar 2.19** STampilan Latihan While

## #LATIHAN 11. WHILE

Buatlah aplikasi kedua untuk pernyataan while dibawah ini pada jendela editor matlab

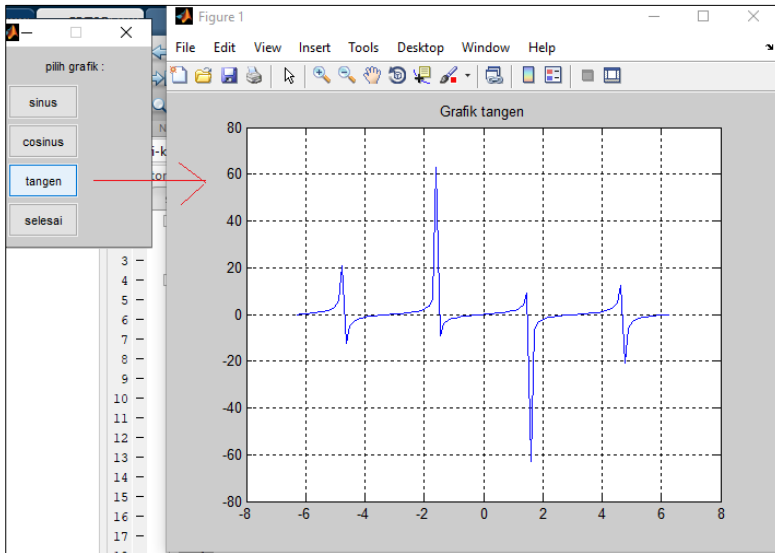
```
function trigono()
x=linspace(-2*pi,2*pi);
con=1;
while con == 1
    k=menu('pilih grafik :','sinus','cosinus','tangen','selesai');
    if k == 4
        con = 0;
    end;
    if con == 1
        if k == 1
            plot(x,sin(x)); grid; title('Grafik sinus');
        elseif k == 2
            plot(x,cos(x)); grid; title('Grafik cosinus');
        elseif k == 3
```

```

        plot(x,tan(x)); grid; title('Grafik tangen');
    end;
end;
end;

```

Output dari aplikasi diatas, akan menampilkan grafik dari fungsi sinus, cosinus dan tangen.



**Gambar 2.20** Tampilan Fungsi Grafik

## F. Graphical User Interface Builder (GUIDE) MATLAB

GUIDE atau GUI builder merupakan sebuah graphical user interface (GUI) yang dibangun dengan obyek grafik seperti tombol (button), kotak teks, slider, Menu dan lain-lain (MathWorks, 2011), aplikasi yang menggunakan GUI umumnya lebih mudah dipelajari dan digunakan karena orang yang menjalankannya tidak perlu mengetahui perintah yang ada dan bagaimana kerjanya.

Jika kita membicarakan pemrograman berorientasi visual, yang ada di benak kita adalah sederetan bahasa pemrograman (Rani, 2013), seperti visual basic, Delphi, visual C++, visual Fox Pro, dan lainnya yang memang didesai secara khusus untuk itu.

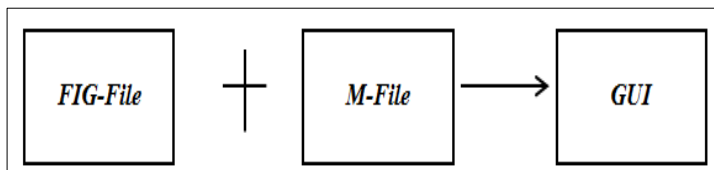


Matlab merintis ke arah pemrograman yang menggunakan GUI dimulai dari versi 5, yang terus disempurnakan sampai sekarang.

GUIDE Matlab mempunyai kelebihan tersendiri dibandingkan dengan bahasa pemrogram lainnya (Sugiharto, 2006), diantaranya:

1. GUIDE Matlab banyak digunakan dan cocok untuk aplikasi-aplikasi berorientasi sains, sehingga banyak peneliti dan mahasiswa menggunakan GUIDE Matlab untuk menyelesaikan riset atau tugas akhirnya.
2. GUIDE Matlab mempunyai fungsi built-in yang siap digunakan dan pemakai tidak perlu repot membuatnya sendiri.
3. Ukuran file, baik FIG-file maupun M-file, yang dihasilkan relatif kecil.
4. Kemampuan grafisnya cukup kita 1 dan tidak kalah dibandingkan dengan bahasa pemrograman lainnya.

Secara sederhana sebenarnya GUI matlab dapat dibentuk oleh dua file (MathWorks, 2011), yaitu Fig-File dan M-file yang adapat digambarkan sebagai berikut:



**Gambar 2.21** Struktur File Matlab

Berdasarkan gambar diatas, dapat dijelaskan sebagai berikut:

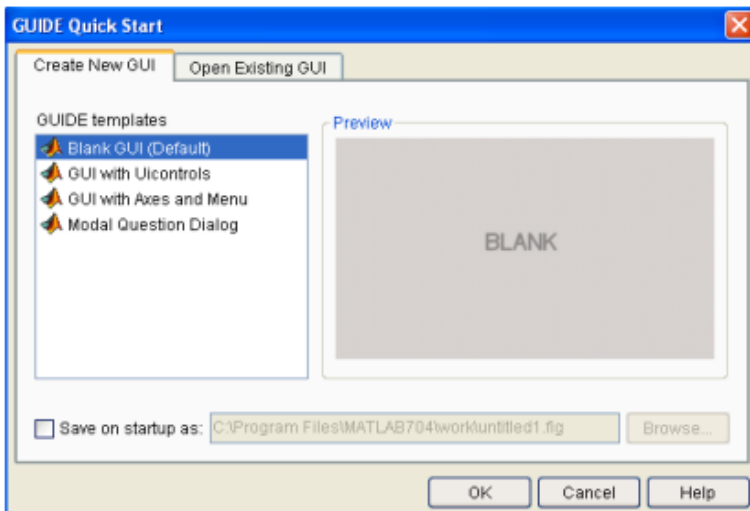
1. File dengan tipe Fig(\*.fig), merupakan file figure yang berisi debuku figure sebuah GUI dengan semua komponen turunanya, seperti nilai-nilai objek yang dapat ditentukan melalui properti
2. File dengan tipe M (\*.M), merupakan file yang berisi source code ataupun fungsi matlab, kontrol GUI, dan callback yang didefinisikan sebagai sebuah subfungsi.

## Mengenal lingkungan Kerja GUI

Untuk masuk kedalam GUI pada matlab, dilakukan dengan menggunakan sintaks dasar yang diketikan pada command windows.

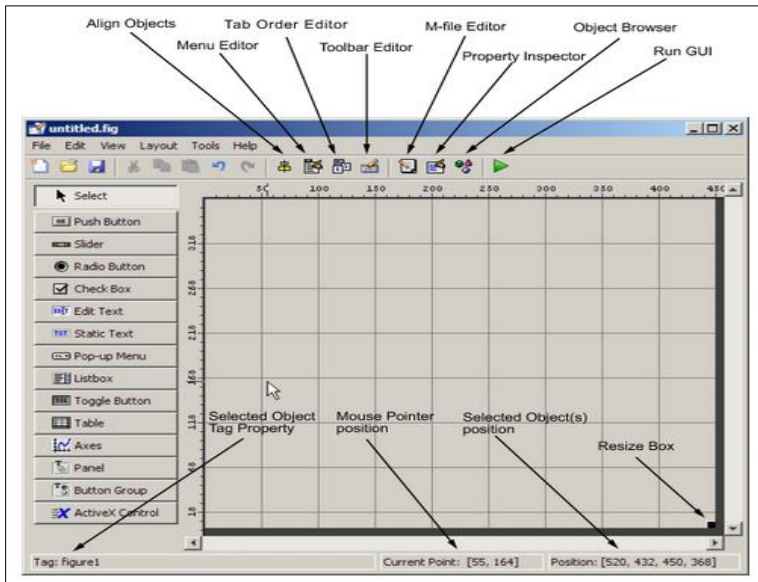


Selanjutnya akan muncul tampilan kotak dialog pilihan GUIDE Quick Start.



**Gambar 2.22** Jendela GUIDE Quick Start

GUIDE Quick Start memiliki dua buah pilihan, yaitu Create New GUI dan Open Existing GUI. Create New GUI digunakan jika kita memang belum pernah membuat aplikasi GUI Matlab atau jika kita memang ingin membuat sebuah figure baru, sedangkan Open Existing GUI digunakan jika kita sudah memiliki file figure matlab atau akan memodifikasi file figure yang telah kita simpan.



**Gambar 2.23** Lingkungan Komponen GUIDE

Semua kontrol pada GUIDE dapat dimunculkan pada layout/figure dengan cara mendrag kiri kontrol yang diinginkan ke figure. Adapun penjelasan fungsi masing-masing kontrol adalah sebagai berikut:

1. Select

Merupakan sebuah komponen kontrol yang berfungsi untuk menseleksi atau memilih dan mengaktifkan komponen yang ada pada area form GUI.

2. Pushbutton

Pushbutton merupakan jenis kontrol berupa tombol tekan yang akan menghasilkan tindakan jika diklik, misalnya tombol OK, Cancel, Hitung, Hapus, dan sebagainya. Untuk menampilkan tulisan pada pushbutton kita dapat mengaturnya melalui property inspector dengan mengklik obyek pushbutton pada figure, lalu mengklik toolbar property inspector atau menggunakan klik kanan lalu pilih property inspector. Selanjutnya isilah tab string dengan label yang diinginkan, misalnya Hitung.

### 3. Toggle Button

Toggle button memiliki fungsi yang sama dengan pushbutton. Perbedaannya adalah saat pushbutton ditekan, maka tombol akan kembali pada posisi semula jika tombol mouse dilepas, sedangkan pada toggle button, tombol tidak akan kembali ke posisi semula, kecuali kita menekannya kembali.

### 4. Radio Button

Radio button digunakan untuk memilih atau menketik satu pilihan dari beberapa pilihan yang ada. Misalnya, sewaktu kita membuat aplikasi operasi Matematika (penjumlahan, pengurangan, perkalian, dan pembagian).

### 5. Edit Text dan Static Text Edit text

Komponen ini digunakan untuk memasukkan atau memodifikasi suatu text yang diinputkan dari keyboard, sedangkan static text hanya berguna untuk menampilkan text/tulisan, sehingga kita tidak bisa memodifikasi/mengedit text tersebut kecuali melalui property inspector. Contoh edit dan static text dapat dilihat pada Gambar 6. 5)

### 6. Panel

Panel merupakan kotak tertutup yang dapat digunakan untuk mengelompokkan kontrol-kontrol yang berhubungan. Tidak seperti kontrol lainnya, frames tidak memiliki rutin callback.

### 7. Checkboxes

Kontrol checkboxes berguna jika kita menyediakan beberapa pilihan mandiri atau tidak bergantung dengan pilihan-pilihan lainnya. Contoh aplikasi penggunaan checkboxes adalah ketika kita diminta untuk memilih hobi. Karena hobi bisa lebih dari satu, maka kita dapat mengklik checkboxes lebih dari satu kali

### 8. Slider

Slider berguna jika kita menginginkan inputan nilai tidak menggunakan keyboard, tetapi hanya dengan cara menggeser slider secara vertical maupun horizontal ke nilai

yang kita inginkan. Dengan menggunakan slider, kita lebih fleksibel dalam melakukan pemasukan nilai data karena kita dapat mengatur sendiri nilai max, nilai min, serta sliderstep.

#### 9. Popup Menu

Pop Menu berguna menampilkan daftar pilihan yang didefinisikan pada String Property ketika mengklik kita panah pada aplikasi dan memiliki fungsi yang sama seperti radio button. Ketika tidak dibuka, popup Menu hanya menampilkan satu item yang menjadi pilihan pertama pada String Property. Popuptmenu sangat bermanfaat ketika kita ingin memberi sebuah pilihan tanpa jarak, tidak seperti radiobutton.

#### 10. Axes

Axes berguna untuk menampilkan sebuah grafik atau gambar (image). Axes sebenarnya tidak masuk dalam UIControl, tetapi axes dapat diprogram agar pemakai dapat berinteraksi dengan axes dan obyek grafik yang ditampilkan melalui axes.

### **Membuat Aplikasi GUI**

MATLAB mengimplementasikan GUI sebagai sebuah figure yang berisi berbagai style obyek UIControl. Selanjutnya kita harus memprogram masing-masing obyek agar dapat bekerja ketika diaktifkan oleh pemakai GUI.

Langkah dasar yang harus dikerjakan dalam membuat GUI adalah :

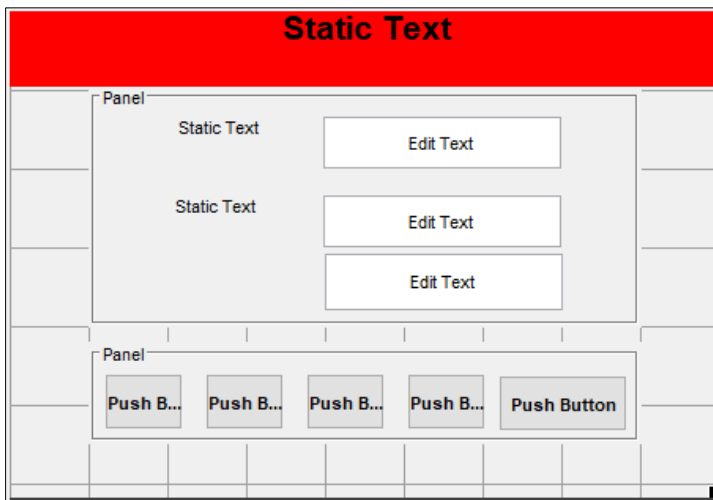
1. Mengatur layout komponen GUI Setelah kita membuka GUIDE Matlab dan telah menentukan template GUI, langkah selanjutnya adalah mendesain figure dengan menggunakan komponen palet sesuai dengan kebutuhan, seperti pushbutton, radiobutton, checkbox, edit text, static text, slider, frames, popup menu, axes, dan sebagainya. Selanjutnya kita dapat mengatur layout masing-masing komponen, baik string(caption), font, color, size, dan sebagainya menggunakan property inspector. Jika kita telah selesai mendesain, jangan lupa untuk menyimpan file figure

yang secara default akan memiliki ekstensi \*.fig. Dari sini, matlab secara otomatis akan membuat sebuah m-file dengan nama yang sama, yaitu file berekstensi \*.m.

2. Memprogram Komponen GUI M-file yang telah dibuat pada langkah sebelumnya, akan otomatis terbuka dan kita harus menulis programnya agar komponen kontrol dapat bekerja secara simultan. Untuk membuat program dalam m-file kita cukup memperhatikan fungsi-fungsi matlab bertkita callback dimana perintah disispkan. Dari langah-langkah dasar diatas, secara sederhana sebenarnya GUI Matlab dibentuk oleh dua buah file, yaitu fig-file dan m-file.





### Aplikasi Matematika Sederhana

Pada GUI builder, **klik file lalu pilih New**, untuk membuat sebuah GUI baru, kemudian simpan dengan nama matematika\_sederhana, setelah itu masukan beberapa kompoenen kedalam form yang masih kosong tersebut, sesuai dengan desain di bawah ini



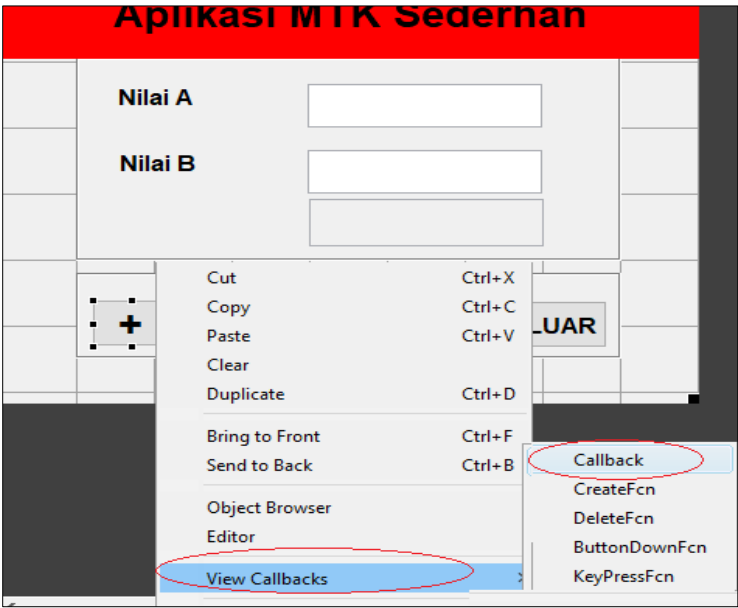
**Gambar 2.24** Rancangan Aplikasi Matematika Sederhana

Pada tabel dibawah ini berisi tentang komponen-komponen, fungsi dan pengaturan properti yang digunakan pada form aplikasi matematika.

ICON	NAMA KOMPONEN	Tag	PROPERTY INSPECTOR
 Panel	1. Panel 1 2. Panel 2	1. Default 2. Default	1. Backgroundcolor=cyan, string=kosongkan 2. Backgroundcolor=blue, string=kosongkan
 StaticText	1. statictext 1 2. statictext 2 3. statictext 3	1. Default 2. Default 3. Default	1. Backgroundcolor=red, string= Aplikasi Matematika, fontsize=20, fontweight = bold 2. string= nilai A, fontsize=12, fontweight = bold 3. string= nilai B, fontsize=12, fontweight = bold
 editText	1. edittext 1 2. edittext 2 3. edittext 3	1. A 2. B 3. hasil	1. tag=A, string =kosongkan 2. tag=B, string = kosongkan 3. tag=hasil, string = kosongkan, enable=off
 PushButton	1. button 1 2. button 2 3. button 3 4. button 4 5. button 5	1. tambah 2. kurang 3. kali 4. bagi 5. keluar	1. string= Tambah, fontsize=12, fontweight = bold 2. sting= Kurang, fontsize=12, fontweight = bold 3. string= Kali, fontsize=12, fontweight = bold 4. string= Bagi, fontsize=12, fontweight = bold

ICON	NAMA KOMPONEN	Tag	PROPERTY INSPECTOR
			5. string= Keluar, fontsize=12, fontweight = bold

Perhatikan pada kolom Tag. Tag merupakan nama untuk komponen yang kita gunakan, dan biasanya akan kita panggil didalam sebuah fungsi, ketika kita butuhkan. Setelah desain rancang aplikasi matematika sederhana selesai, langkah selanjutnya adalah memasukan sourcecode kedalam komponen-komponen button sesuai dengan fungsinya masing-masing. Klik kanan pada tombol + kemudian pilih view callback/callback, untuk masuk kedalam fungsi dari tombol tambah tersebut.





Setelah muncul jendela editor dan berada pada fungsi button tambah, silahkan masukan source code di bawah ini.

```
function tambah_Callback(hObject, eventdata, handles)
nilai_a=str2num(get(handles.a,'string'));
nilai_b=str2num(get(handles.b,'string'));
tambah=nilai_a+nilai_b;
set(handles.hasil,'string',tambah);
```

**Penjelasan code:**

Nilai\_a/b= variabel yang berfungsi menampung input yang ada

di inputkancpengguna pada edit text.

get(handles= mengambil suatu nilai dari komponen GUI

Str2num= fungsi untuk mengubah tipe data string menjadi tipe data number

get(handles= mengambil suatu nilai dari komponen GUI

set(handles= menempatkan suatu nilai kedalam komponen GUI

Lakukanlah hal yang sama untuk tombol, kurang, kali dan bagi, masukan sourcecode sesuai dengan fungsinya masing-masing.

```
function kurang_Callback(hObject, eventdata, handles)
nilai_a=str2num(get(handles.a,'string'));
nilai_b=str2num(get(handles.b,'string'));
tambah=nilai_a-nilai_b;
set(handles.hasil,'string',tambah);
```

% --- Executes on button press in kali.

```
function kali_Callback(hObject, eventdata, handles)
nilai_a=str2num(get(handles.a,'string'));
nilai_b=str2num(get(handles.b,'string'));
tambah=nilai_a*nilai_b;
set(handles.hasil,'string',tambah);
```

% --- Executes on button press in bagi.

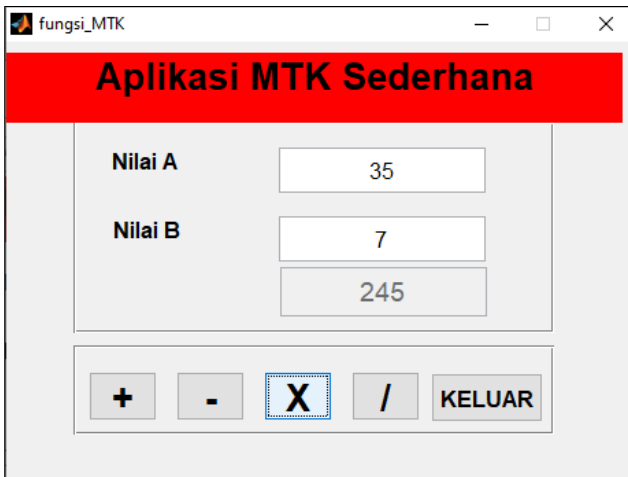
```
function bagi_Callback(hObject, eventdata, handles)
```

```

nilai_a=str2num(get(handles.a,'string'));
nilai_b=str2num(get(handles.b,'string'));
tambah=nilai_a/nilai_b;
set(handles.hasil,'string',tambah);

```

Silahkan jalankan aplikasi dan lakukan pengujian untuk setiap fungsi dari tombol-tombol tersebut, dan pastikan semua telah berfungsi dengan baik.



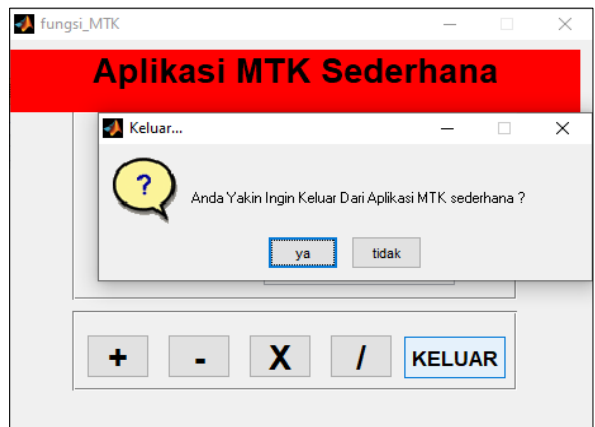
Langkah terakhir adalah memasukan source code untuk fungsi tombol keluar, dimana pada tombol ini terdapat bentuk penggunaan if-end sederhana.

```

function keluar_Callback(hObject, eventdata, handles)
selection=questdlg(['Kita Yakin Ingin Keluar Dari Aplikasi MTK sederhana','?'],...
    ['Keluar','...'],...
    'ya','tidak','ya');
if strcmp(selection,'tidak')
    return;
end
close;

```

Lakukanlah pengujian fungsi untuk tombol keluar yang baru saja kita masukan.



**Gambar 2.25** Tampilan Aplikaasi MTK sederhana




### Aplikasi Penilaian Mahasiswa


Buatlah desain rancangan aplikasi baru, kemudian masukan komponen-komponen GUI yang di perlukan kedalam area form tersebut, adapun desain untuk aplikasi ini, seperti gambar dibawah ini.

APLIKASI NILAI MAHASISWA		
NAMA	<input type="text"/>	<b>NILAI AKHIR</b>  <input type="text"/>  <b>GRADE</b>  <input type="text"/>
NIM	<input type="text"/>	
NILAI TUGAS	<input type="text"/>	
NILAI UTS	<input type="text"/>	
NILAI UAS	<input type="text"/>	
<b>HITUNG</b> <b>BARU</b> <b>KELUAR</b>		

**Gambar 2.26** Rancangan Aplikasi Nilai Mahasiswa

Berdasarkan gambar rancangan form aplikasi perhitungan nilai diatas, komponen-komponen yang digunakan adalah sebagai berikut:

ICON	NAMA KOMPONEN	Tag	PROPERTY INSPECTOR
 Panel	<ol style="list-style-type: none"> <li>Panel 1</li> <li>Panel 2</li> <li>Panel 3</li> </ol>	<ol style="list-style-type: none"> <li>Default</li> <li>Default</li> <li>Default</li> </ol>	<b>string=kosongkan</b>
 StaticText	<ol style="list-style-type: none"> <li>statictext 1</li> <li>statictext 2</li> <li>statictext 3</li> <li>statictext 4</li> <li>statictext 5</li> <li>statictext 6</li> <li>statictext 7</li> <li>statictext 8</li> </ol>	Semua default	<ol style="list-style-type: none"> <li><b>Backgrouncolor=red, string= Aplikasi Nilai Mahasiswa, fontsize=20, fontweight = bold</b></li> <li><b>string= Nama A, fontsize=12, fontweight = bold</b></li> <li><b>string= Nim, fontsize=12, fontweight = bold</b></li> <li><b>string= Tugas, fontsize=12, fontweight = bold</b></li> <li><b>string= UTS, fontsize=12, fontweight = bold</b></li> <li><b>string= UAS, fontsize=12, fontweight = bold</b></li> <li><b>string= Nilai Akhir, fontsize=12, fontweight = bold</b></li> <li><b>string= Grade, fontsize=12, fontweight = bold</b></li> </ol>
 editText	<ol style="list-style-type: none"> <li>edittext 1</li> <li>edittext 2</li> <li>edittext 3</li> <li>edittext 4</li> </ol>	<ol style="list-style-type: none"> <li>nama</li> <li>nim</li> <li>tugas</li> <li>uts</li> </ol>	<ol style="list-style-type: none"> <li>✓ poin 1-5 semua string =kosongkan</li> <li>✓ poin 6-7 (hasil &amp; grade) string = kosongkan, enable=off , backgrouncolor=cyan, fontcolor=red, fontweight = bold</li> </ol>

	5. editText 5 6. editText 6 7. editText 7	5. uas 6. hasil 7. grade	
 PushButton	1. button 1 2. button 2 3. button 3	1. hitung 2. baru 3. keluar	<b>1. string= Hitung, fontsize=12, fontweight = bold</b> <b>2. sting= Baru, fontsize=12, fontweight = bold</b> <b>3. string= Keluar, fontsize=12, fontweight = bold</b>

Setelah desain rancangan form selesai di buat, langkah selanjutnya adalah memasukan source code. Pada tombol hitung, silahkan kita klik kanan kemudian pilih viewcallback/ callback, kemudian masukan source code dibawah ini.

```
function hitung_Callback(hObject, eventdata, handles)
    tugas=str2num(get(handles.tugas,'string'));
    uts=str2num(get(handles.uts,'string'));
    uas=str2num(get(handles.uas,'string'));
    na=(tugas*20/100)+(uts*30/100)+(uas*50/100);
    set(handles.hasil,'string',na);
    if na>80 && na<=100
        set(handles.grade,'string','A');
    elseif na<=80 && na>70
        set(handles.grade,'string','B');
    elseif na<=70 && na>60
        set(handles.grade,'string','C');
    elseif na<=60 && na>50
        set(handles.grade,'string','D');
    else
        set(handles.grade,'string','E');
    end
```

**Penjelasan code;**

**str2num :** berfungsi untuk mengubah nilai string menjadi numerik

**(get(handles.... :** berfungsi untuk mengambil nilai dari sebuah komponen didalam form

**(set(handles.... :** berfungsi untuk menempatkan suatu nilai kedalam sebuah komponen didalam form

**na=(tugas\*20/100.... :** variabel yang berfungsi untuk menghitung nilai akhir sesuai rumus yang ditentukan.

**If, elseif, else, end:** control flow yang akan dieksekusi untuk mencari statement atau kondisi yang bernilai benar (true)

Jalankan aplikasi kemudian lakukan pengujian fungsi untuk tombol hitung nilai mahasiswa.

APLIKASI NILAI MAHASISWA	
NAMA	dimas
NIM	202019
NILAI TUGAS	90
NILAI UTS	76
NILAI UAS	65
NILAI AKHIR	
73.3	
GRADE	
B	
HITUNG BARU KELUAR	

**Gambar 2.27** Tampilan Aplikasi Nilai Mahasiswa

Setelah memastikan fungsi hitung sudah tidak terdapat error, lanjutkan memasukan source code untuk tombol baru, yang akan berfungsi untuk membersihkan semua nilai komponen edit text, sehingga pengguna bisa memasukan nilai yang baru. Danjuga untuk tombol keluar yang berfungsi untuk keluar atau metu aplikasi.

```
% --- di eksekusi ketika menekan tombol baru
function baru_Callback(hObject, eventdata, handles)
    proyek2=guidata(gcbo);
    baru="";
    set(proyek2.nama,'string',baru);
    set(proyek2.nim,'string',baru);
    set(proyek2.tugas,'string',baru);
    set(proyek2.uts,'string',baru);
    set(proyek2.uas,'string',baru);
    set(proyek2.hasil,'string',baru);
```

<pre>set(projek2.grade,'string',baru);</pre>
<pre>% --- di eksekusi ketika menekan tombol keluar function keluar_Callback(hObject, eventdata, handles) close;</pre>
<p><b>Penjelasan code:</b></p> <p><b>projek2 :</b> variabel yang berfungsi untuk menampung nilai ketika tombol baru dijalankan</p> <p><b>baru=" :</b> variabel yang berisi tipe data string yang kosong</p> <p><b>close :</b> perintah yang berfungsi untuk menutup form aktif</p>

Jalankan aplikasi dan silahkan uji semua fungsi yang ada pada semua tombol, perhatikan dan amati setiap fungsi dan kita gunakan apakah telah sesuai atau belum.





### Aplikasi Trigonometri




Pada aplikasi ketiga ini kita akan membuat sebuah aplikasi yang akan menampilkan grafik dari fungsi sinus, cosinus dan tangen, dimana pada bagian sebelumnya kita sudah buat dengan menggunakan bentuk m-file. Berikut adalah rancangan form pada aplikasi trigonometri yang akan kita buat.

**Gambar 2.28** Rancangan Aplikasi Trigonometri



Sesuai dengan rancangan form aplikasi, komponen-komponen dan pengaturan properti komponen tersebut dijelaskan pada tabel dibawah ini.

ICON	NAMA KOMPONEN	TAG	PROPERTY INSPECTOR
 Panel	1. Panel 1 2. Panel 2 3. Panel 3 4. Panel 4	Semua default	Semua string dikosongkan
 StaticText	1. Statictext 1 2. Statictext 2 3. Statictext 3 4. Statictext 4	Semua default	1. string= Aplikasi Trigonometri, fontsize=20, fontweight = bold 2. string= Amplitud, fontsize=12, fontweight = bold 3. string= Phase, fontsize=12, fontweight = bold 4. string= Sudut, fontsize=12, fontweight = bold
 editText	1. Edittext 1 2. Edittext 2	1. ampli 2. phase	semua string =kosongkan
 Menu PopUp	Menu PopUp1	sudut	String= 1,2,3,4

ICON	NAMA KOMPONEN	TAG	PROPERTY INSPECTOR
 Radio Button	1. Radio button 1 2. Radio button 2 3. Radio button 3	1. si 2. cs 3. tn	1. <b>string= Sin, fontsize=12, fontweight = bold</b> 2. <b>string= Cos, fontsize=12, fontweight = bold</b> 3. <b>string= Tan, fontsize=12, fontweight = bold</b>
 Axes	Axes1	Axes1	<b>default</b>
 PushButton	1. Button 1 2. Button 2 3. Button 3	1. Proses 2. Bersih 3. Keluar	4. <b>string= Proses, fontsize=12, fontweight = bold</b> 5. <b>sting= Bersih, fontsize=12, fontweight = bold</b> 6. <b>string= Keluar, fontsize=12, fontweight = bold</b>

Tahap berikutnya silahkan kita masukan fungsi pada setiap radiobutton, dengan cara mengklik kanan/viewcallback/callback, perhatikan tag setiap komponen radio button tersebut, dan masukan fungsi sesuai dengan tag radio button tersebut.

```
% --- perintah di Eksekusi ketika radio button sin ditekan
function si_Callback(hObject, eventdata, handles)
set(handles.cs,'Value',0);
set(handles.tn,'Value',0);
pilih = 1;
handles.pilih=pilih;
guidata(hObject,handles)
```

```
% --- perintah di Eksekusi ketika radio button cos ditekan
function cs_Callback(hObject, eventdata, handles)
set(handles.tn,'Value',0);
set(handles.si,'Value',0);
pilih = 2;
handles.pilih=pilih;
guidata(hObject,handles)
```

```
% --- perintah di Eksekusi ketika radio button tan ditekan
function tn_Callback(hObject, eventdata, handles)
set(handles.cs,'Value',0);
set(handles.si,'Value',0);
pilih = 3;
handles.pilih=pilih;
guidata(hObject,handles)
```

#### **Penjelasan code:**

**Pilih :** variabel yang digunakan untuk menampung nilai, dimana ketika radiobutton sin di tekan maka bernilai 1, cos bernilai 2 dan tangen bernilai 3

**Value, 0 :** mengatur nilai 0, untuk dua radiobutton yang lain, ketika kondisi salah satu radio button di tekan

Langkah selanjutnya adalah memasukan fungsi callback untuk komponen Menu PopUp sudut, masukan sourcecode dibawah ini.

```
% --- perintah di Eksekusi ketika Menu popup dipilih
function sudut_Callback(hObject, eventdata, handles)
sudut=get(hObject,'value');
    switch sudut
        case 1
            valuesudut=1;
        case 2
            valuesudut=2;
        case 3
            valuesudut=3;
        case 4
            valuesudut=4;
    end
handles.valuesudut= valuesudut;
guidata(hObject, handles);
```

**Penjelasan code:**

**Sudut= : variabel yang digunakan untuk menampung nilai dari pilihan dari Menu PopUp**

**Switch-Case : fungsi yang digunakan untuk menguji setiap kasus sampai salah satu kasus benar.**

Bagian utama yang akan melakukan perhitungan dan menampilkan grafik kedalam komponen axes adalah tombol proses. Silahkan masukan source code berikut, perhatikan dengan teliti pemanggilan tag sebagai nama dari komponen didalam area form yang akan di panggil.

```
% --- di eksekusi ketika menekan tombol proses
function proses_Callback(hObject, eventdata, handles)
sudut=handles.valuesudut;
A=str2num(get(handles.ampli,'string'));
r=str2num(get(handles.phase,'string'));
```

```

x=-10:0.1:10; pilih=handles.pilih
switch handles.pilih
    case 1
        y=A*sin(sudut*x+r);
    case 2
        y=A*cos(sudut*x+r);
    case 3
        y=A*tan(sudut*x+r);
end
axes(handles.axes1);
a=0; plot(x,y,'r', a, x, 'b', x, a, 'b');
grid on

```

**Penjelasan code:**

**Sudut:** variabel yang mengambil nilai sudut dari Menu popup

**A=:** variabel yang mengambil nilai amplitudo sudut dari edit text1

**r=:** variabel yang mengambil nilai phase dari edit text1

**str2num:** fungsi untuk mengkonversi tipe data string menjadi numeric

**x=-10:0.1:10;** %membangkitkan sumbu x

**pilih= :** variabel yang mengambil nilai pilihan pada radiobutton

Masukan juga fungsi callback pada tombol bersih yang digunakan untuk membersihkan komponen-komponen edit teks dan gambar pada komponen axes, yang digunakan jika ingin melakukan proses yang baru.

```

% --- di eksekusi ketika menekan tombol bersih.
function bersih_Callback(hObject, eventdata, handles)
set(handles.ampli, 'string', '');
set(handles.phase, 'string', '');
set(handles.cs, 'Value', 0);
set(handles.si, 'Value', 0);
set(handles.tan, 'Value', 0);

```

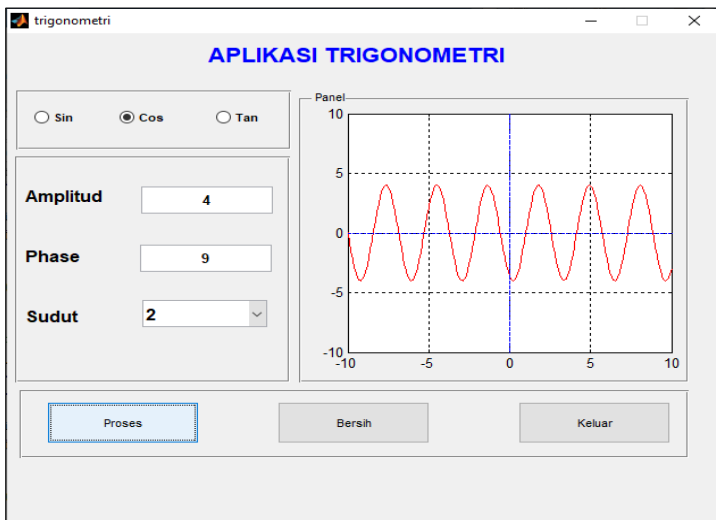
```

axes(handles.axes1);
plot(0);

% --- di eksekusi ketika menekan tombol keluar
function keluar_Callback(hObject, eventdata, handles)
close;

```

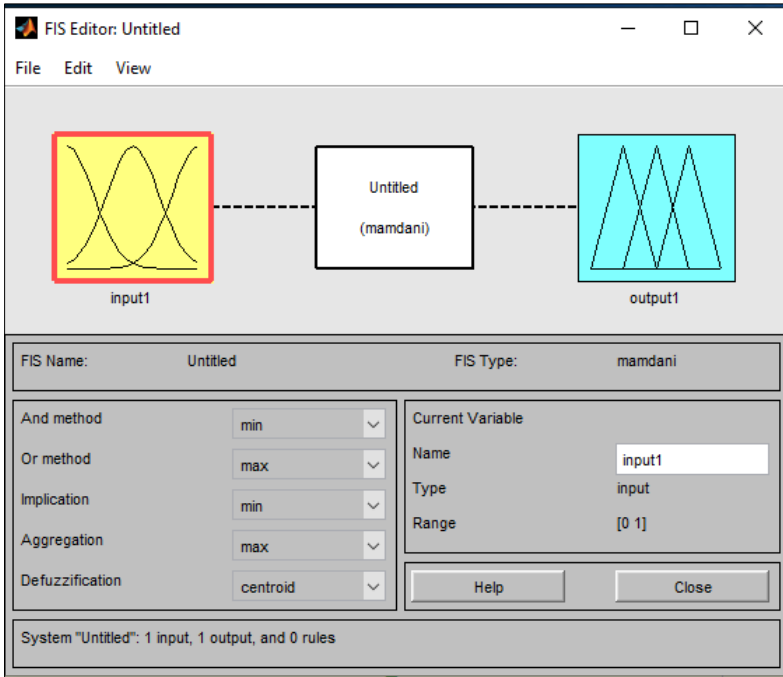
Running atau jalan kan aplikasi dengan menekan tombol f5 dan uji semua fungsi proses perhitungan dan pembuatan grafik.



**Gambar 2.29** Tampilan Aplikasi Trigonometri

## G. Fuzzy Logic Toolbox

Matlab menyediakan tools atau alat untuk melakukan analisis, merancang, dan mensimulasikan sistem logika fuzzy. Dengan tools ini kita dapat menentukan dan mengonfigurasi input, output, fungsi keanggotaan, dan aturan sistem inferensi fuzzy. Toolbox memungkinkan kita secara otomatis menyetel fungsi keanggotaan dan aturan sistem inferensi fuzzy dari data. Kita dapat mengevaluasi sistem logika fuzzy yang dirancang di MATLAB dan Simulink.



**Gambar 2.30** Tampilan Fuzzy Logic Toolbox

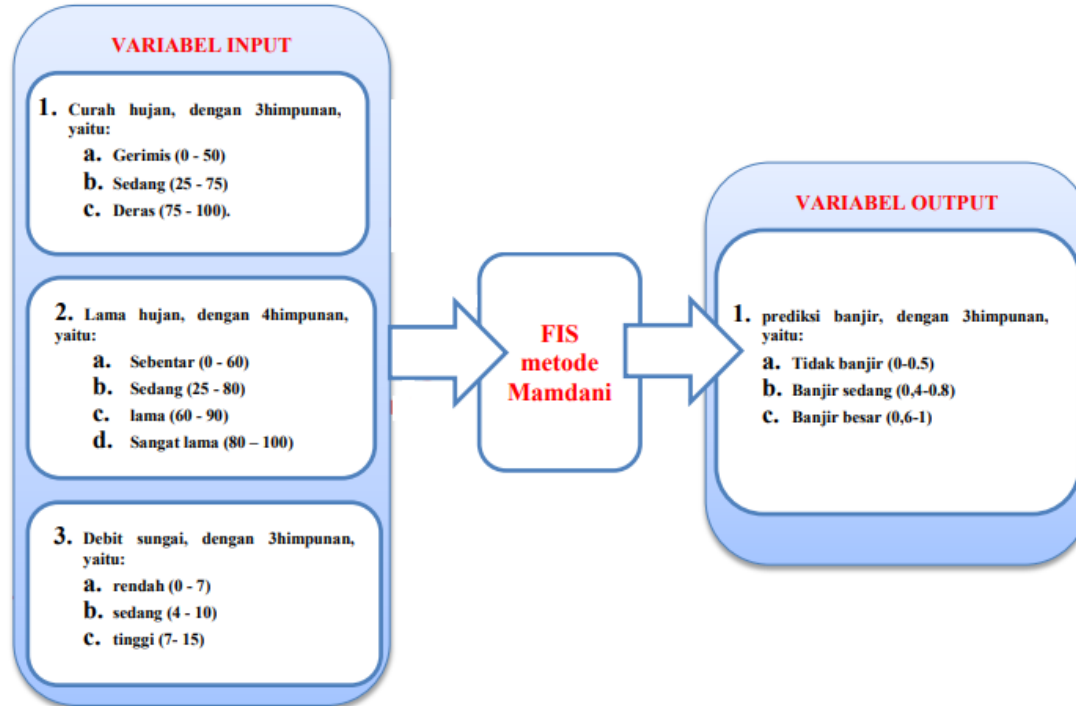
Selain itu, sistem inferensi fuzzy dapat digunakan sebagai sistem pendukung untuk menjelaskan model black-box berbasis kecerdasan buatan (AI). Kita dapat membuat aplikasi yang berbasis GUI dan mengintegrasikannya dengan model sistem fuzzy yang telah kita buat, bahkan kita dapat menjadikannya aplikasi atau file executable mandiri.

Hal yang paling penting untuk kita sadari adalah pemahaman kita secara teoritis tentang konsep dari logika fuzzy itu sendiri, karena seanggih dan sehebat apapun tools yang kita gunakan hanya bersifat sebagai alat bantu, yang akan digunakan untuk melakukan proses dari rancangan sistem fuzzy yang telah kita buat. Dalam artian ini maka kita harus benar-benar memahami dan teliti terkait dengan kasus apa yang akan kita selesaikan dengan sistem fuzzy dan juga memperhatikan desain dan rancangan sistem fuzzy yang akan kita buat sebelum menggunakan atau dimasukkan ke dalam toolbox tersebut.

### **Mulai Menggunakan Fuzzy logic Toolbox**

Seperti yang telah saya sebutkan sebelumnya, bahwa dalam membangun sistem fuzzy adalah seberapa jauh seorang pembuat sistem memahami dan mengerti tentang sistem yang akan ia buat ( berkenaan dengan parameter – parameter yang berpengaruh terhadap sistem ataupun berkenaan dengan masalah – masalah dari sebuah kasus yang ingin diselesaikan dengan sistem fuzzy). Dalam kasus ini, untuk memprediksi datangnya banjir setidaknya dipengaruhi oleh beberapa parameter, yaitu: curah hujan, lamanya hujan dan debit air di sungai. Sehingga kerangka pemikiran yang digunakan untuk membangun sistem prediksi banjir dengan menggunakan FIS metode mamdani dapat dilihat pada gambar dibawah ini:



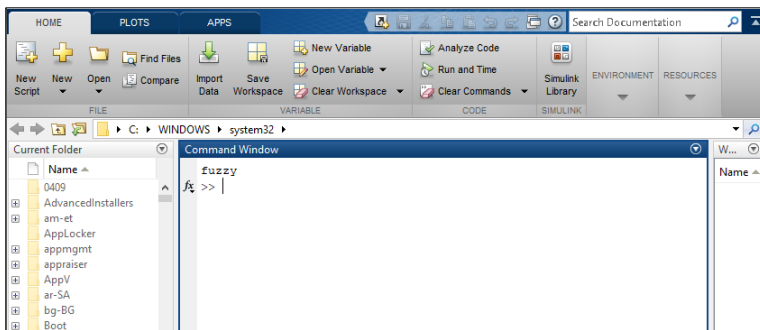


**Gambar 2.31** Rancangan Sistem Fuzzy

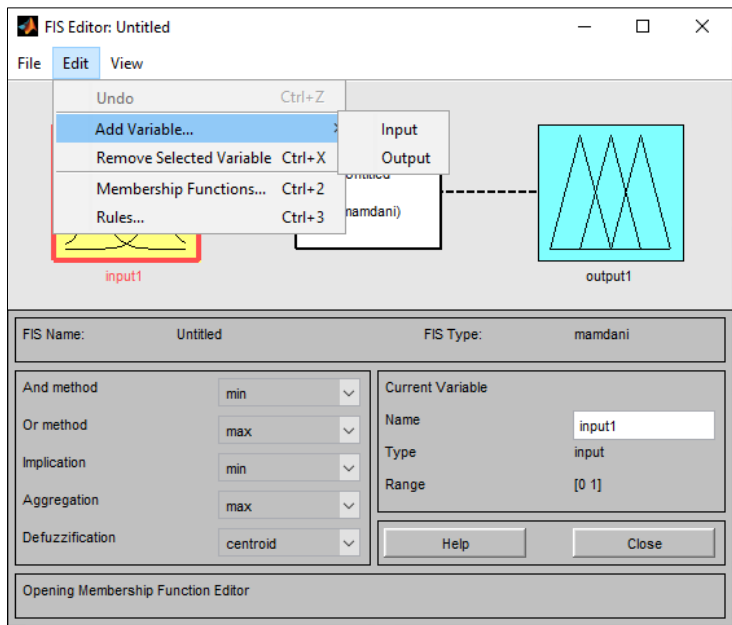
## Desain Dan Rancangan Sistem Fuzzy

Sesuai dengan kerangka pemikiran dan desain dari sistem yang akan dibangun, maka langkah selanjutnya adalah melakukan simulasi dan pembuaatn aplikasi menggunakan fuzzy toolbox software matlab, berikut adalah langkah-langkah pengerjaanya:

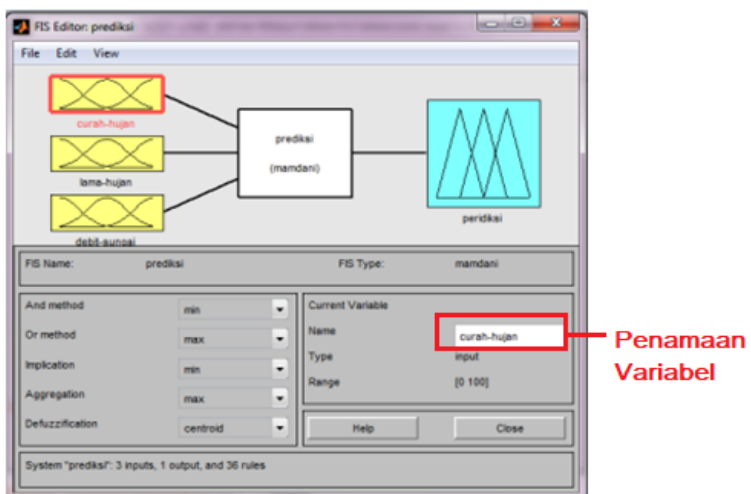
1. Jalankanlah program matlab anda lalu di command window ketik "fuzzy", untuk masuk kedalam toolbox fuzzy



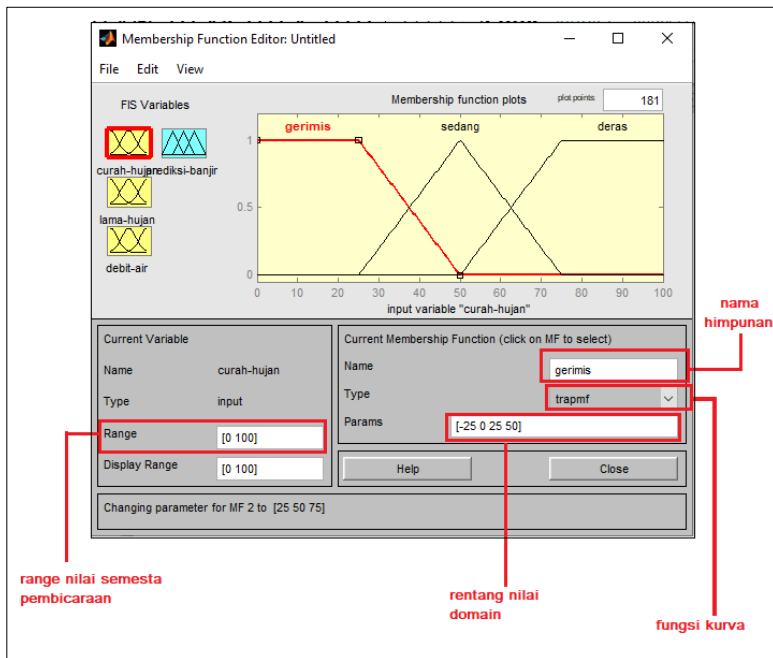
2. Pada FIS editor yang muncul Klik menu edit-add variable-input untuk menentukan banyaknya input yang kita inginkan. Dan klik edit-add variable-output untuk menentukan output yang kita inginkan seperti tampilan dibawah ini



Sesuaikan dengan rancangan pada kerangka pemikiran diatas, pada tab komponen current variabel, masukan nama setiap variabel input dan output dibagian name, sehingga desain dari sistem fuzzy akan terlihat seperti gambar berikut

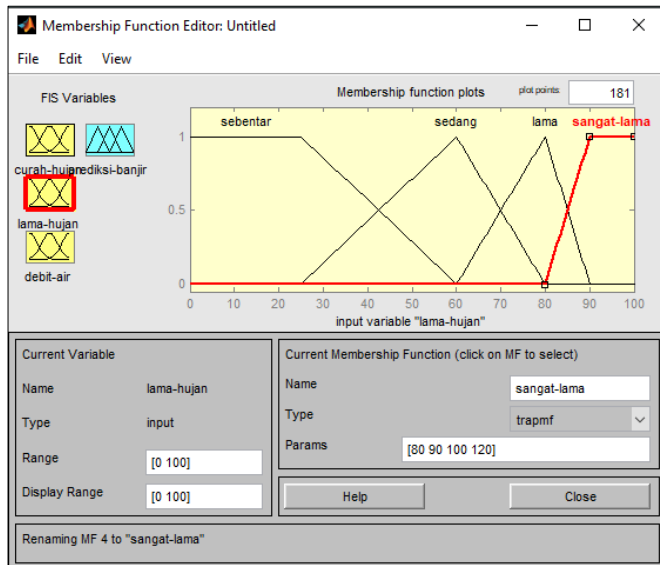


3. Klik curah hujan 2 kali. kemudian masukan range nilai semesta pembicaraan, masukan nama himpunan, fungsi keanggotaan berdasarkan data di atas. untuk kurva trapesium gunakan trapmf kemudian masukan nilai domain untuk himpunan gerimis  $[-25;0;25;50]$ , sedangkan untuk kurva segitiga gunakan trimf dengan nilai domain untuk himpunan sedang  $[25;50;75]$  dan pada himpunan deras menggunakan kurva trapesium dengan menggunakan komponen trmf dengan nilai domain  $[50;75;100;125]$ . Untuk menambahkan garis kurva di membership function editor. klik edit-MFs. Sehingga diperoleh tampilan sebagai berikut:

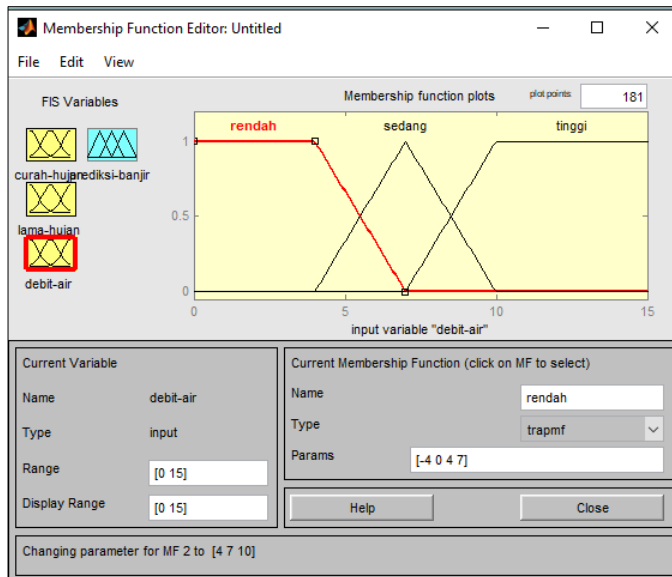


Perhatikan pada kerangka dan desain yang telah dibuat sebelumnya.

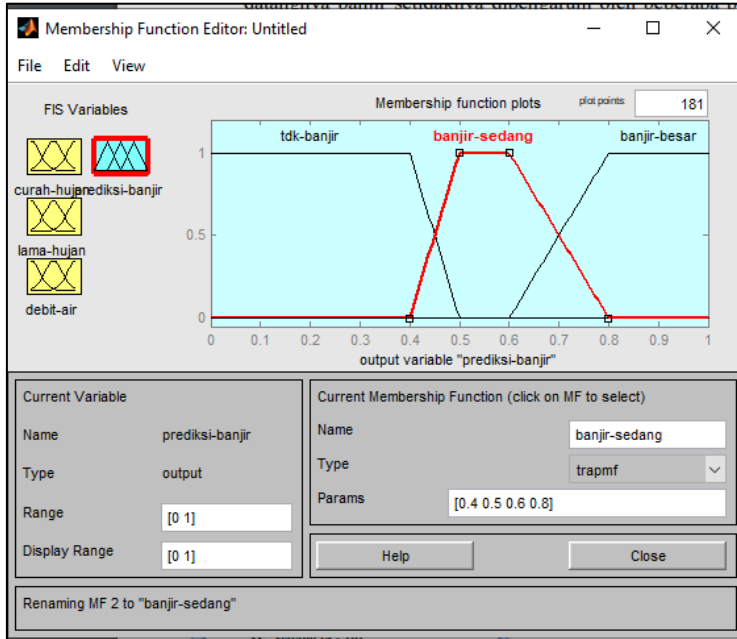
4. Lakukan hal yang sama untuk variabel input lama hujan, sesuaikan dengan rancangan dan desain, sehingga diperoleh tampilan sebagai berikut:



5. Lakukan hal yang sama untuk variabel input debit air, sesuaikan dengan rancangan dan desain sehingga diperoleh tampilan sebagai berikut:



6. Lakukan hal yang sama juga untuk variabel output prediksi banjir. kemudian masukan himpunan fungsi keanggotaan berdasarkan data di atas. Sehingga diperoleh tampilan sebagai berikut :



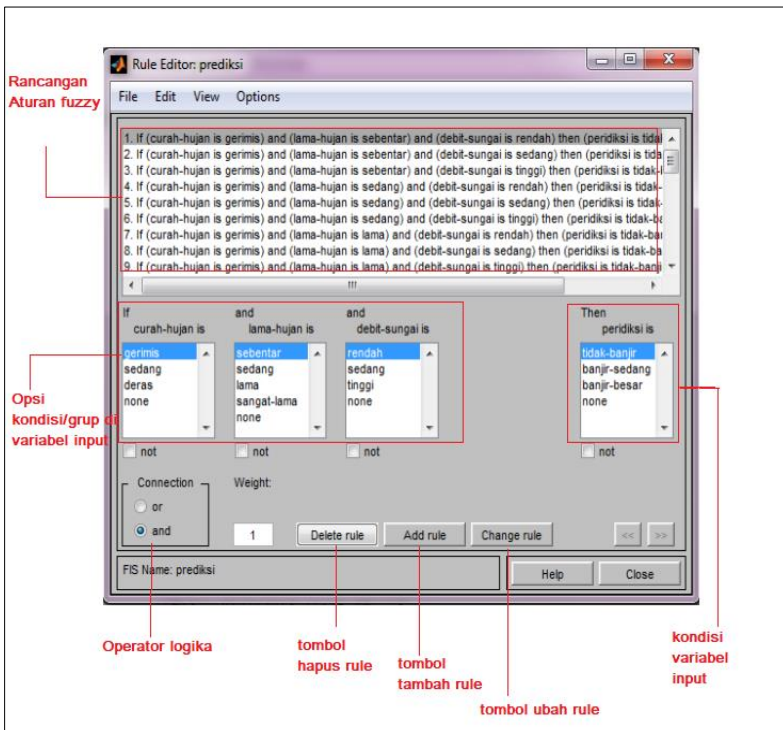
Setelah selesai memasukkan semua variabel input dan output beserta dengan komponen-komponen dari setiap variabel pada editor membership function atau fungsi keanggotaan, maka tahap pertama dalam membangun model sistem fuzzy telah selesai dan kita bisa melanjutkan ke tahap berikutnya.

### Memasukan Aturan Sistem Fuzzy

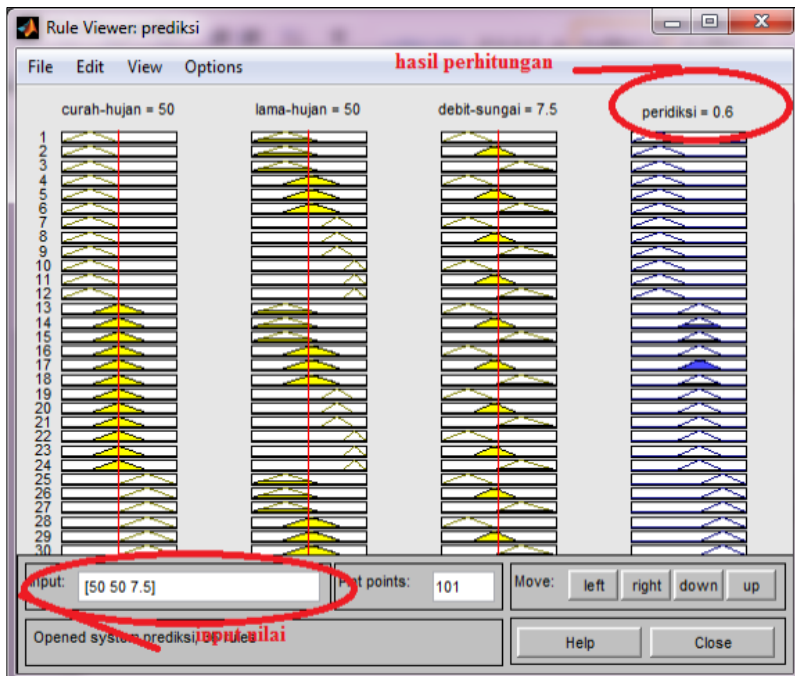
Pada tahap kedua ini kita akan memasukkan basis aturan kedalam sistem fuzzy sebagai basis pengetahuan dari model yang akan kita bangun, sebelum itu pastikan anda telah membuat sebuah rancangan aturan tersebut sebelum dimasukkan kedalam model fuzzy.

Berdasarkan pada kerangka pemikiran sebagai dasar desain dari model sistem fuzzy yang akan kita bangun, maka dapat diketahui bahwa total dari aturan fuzzy yang terbentuk sebanyak 36 aturan dasar dari sistem fuzzy yang dibangun. Adapun langkah-langkah dalam memasukkan aturan fuzzy pada toolbox ini adalah:

1. Pada menu bar, klik edit-Rules dan tulis aturan berdasarkan input dan output yang kita buat. Maka diperoleh tampilan sebagai berikut.

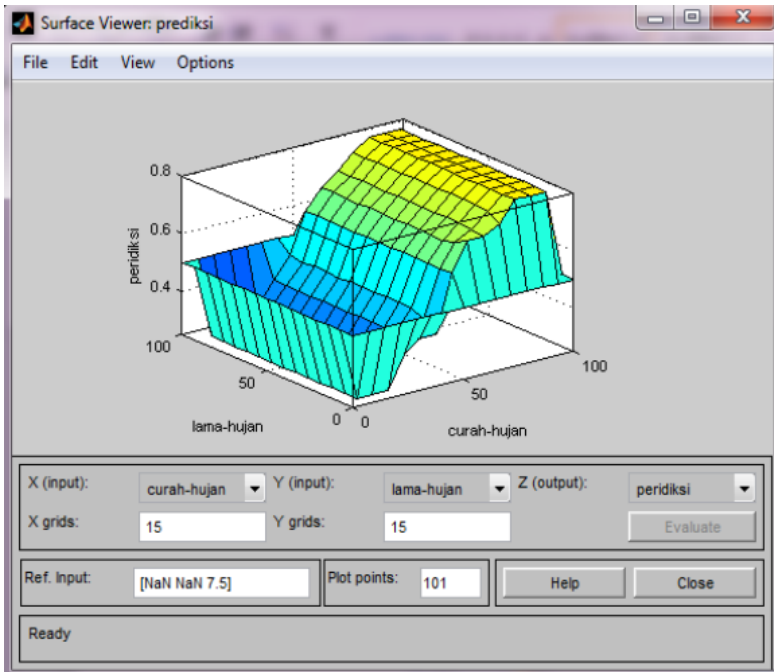


- Setelah proses pembuatan aturan selesai dibuat. kemudian kita klik pada Rule-editor yaitu view-Rules maka muncul tampilan dibawah ini, yang digunakan untuk melakukan prses simulasi dan defuzzifikasi sesuai dengan rule yang telah diberikan

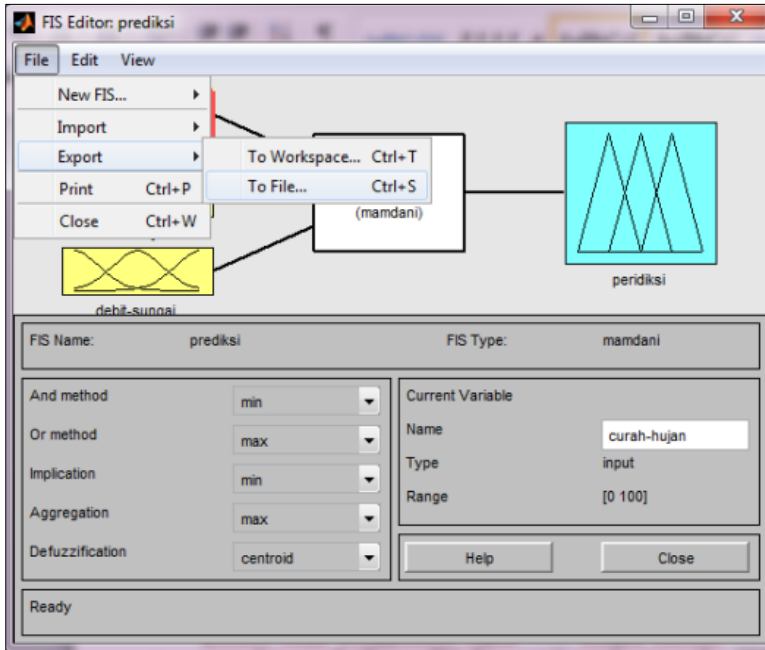




3. Pada editor rule viewer diatas, kita dapat lngsung melihat ouput dari model sistem fuzzy yng telah dibuat, kita dapat langsung memasukan nilai-nilai untuk variabel input kedalam textbox input yang trsedia, maka hasil dari model sistem fuzzy sudah otomatis muncul pada jendela editor tersebut. Jika ingin melihat grafik maka klik menu view-surface pada tampilan Rule Viewer.



- Langkah terakhir adalah menyimpan file sistem fuzzy yang telah anda buat, klik FileExport-To file Seperti tampilan dibawah ini.

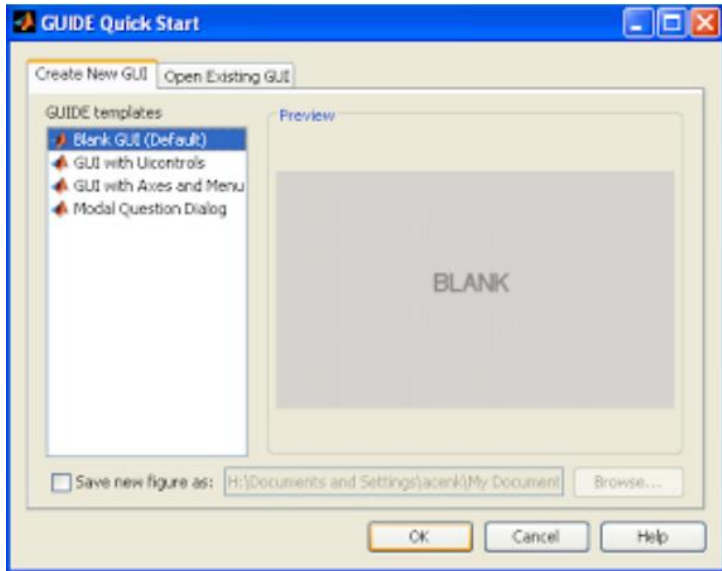


## Membuat Aplikasi GUI

Setelah model sistem fuzzy selesai dibuat dan telah dilakukan proses pengujian model, kita dapat mengintegrasikannya dengan aplikasi berbasis GUI dan juga database. Hal penting yang harus anda ingat adalah bahwa kita harus menyimpan file model prediksi banjir fuzzy dan file GUI dalam satu folder.

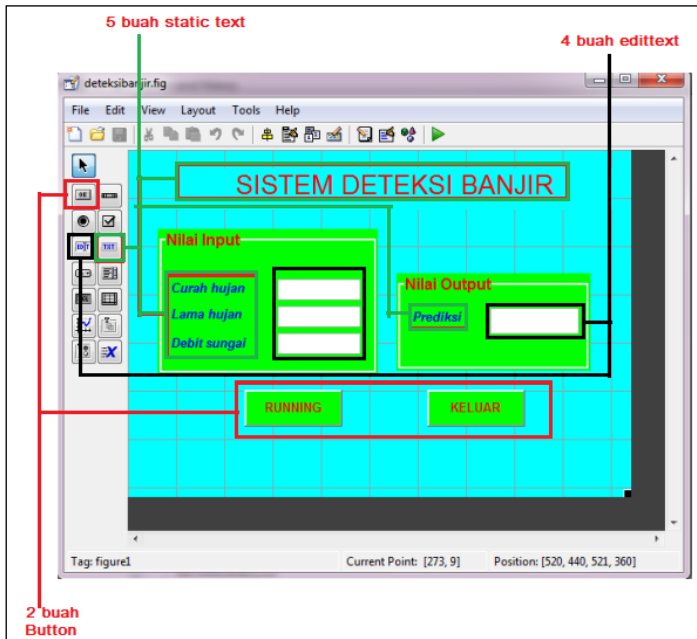
Adapun tahapan yang untuk membuat aplikasi berbasis GUI, adalah proses perancangan antarmuka dari aplikasi dan proses pengkodean atau pemberian source kode untuk semua komponen yang diperlukan, berikut adalah lanka-langkah pembuatannya:

1. Masuk kedalam GUIDE Builder dengan menjalankan perintah guide pada command windows sehingga muncul tampilan sebagai berikut:



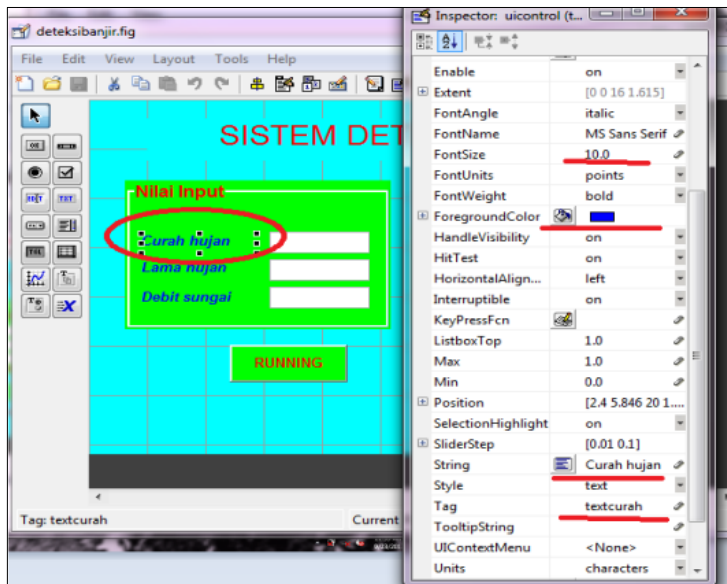
Pada editor di atas pilih blank gui, kemudian simpan file kedalam satu folder dengan file model yang telah kita buat.

2. Tambahkan komponen-komponen yang diperlukan kedalam form yang baru muncul, perhatikan gambar dibawah ini



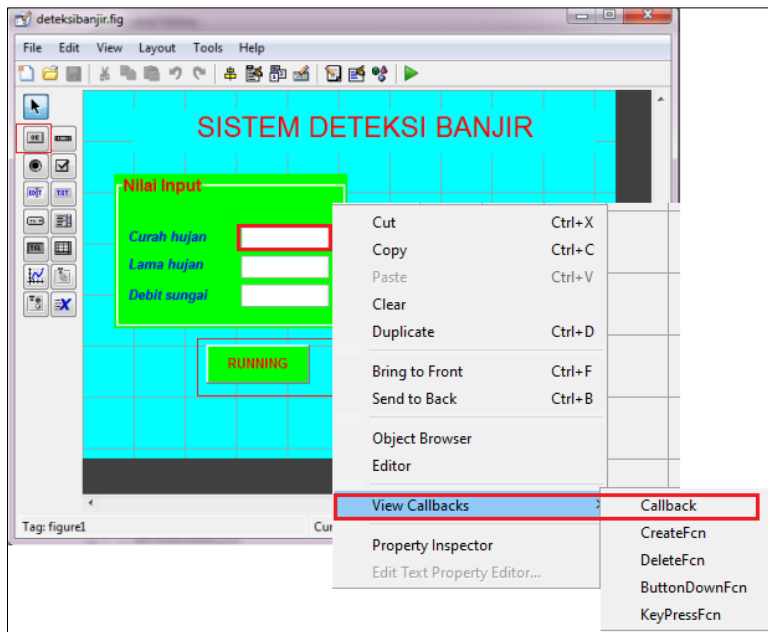
Pada form diatas komponen yang digunakan adalah 2 buah Button Group untuk menampung komponen input dan komponen output, kemudian 2 buah push button, 4 buah edit text dan 5 buah static text.

3. Untuk melakukan pengaturan terhadap tiap - tiap komponen dapat dilakukan dengan cara mengklik tombol property inspector atau bisa juga dengan cara double klik komponen yang bersangkutan.



Yang penting untuk di perhatikan property inspector adalah pada bagian *tag* di setiap komponen karena itu merupakan nama komponen yang akan kita panggil/gunakan, saat memasukan source code. Pada property inspector ubahlah tag name tersebut, untuk komponen edit text yang berfungsi untuk menampung nilai inputan dari variabel input.

Ubahlah Editext1:tag = Chujan, edittext2:tag=lamahuj, edittext3:tag=debit, untuk semua komponen tersebut kosongkanlah pad bagian string, kemudian klik kanan pada komponen editext1 pilih view callback/callback, perhatikan gambar berikut.



Setelah source code editor muncul, masukanlah source kode berikut ini

```
function Chujan_Callback(hObject, eventdata, handles)
Chujan=str2double(get(hObject,'string'));
handles.curah hujan=Chujan;
guidata(hObject, handles);
```

4. Dengan langkah yang sama, masukan juga sourcecode dibawah ini untuk edittext2

```
function lamahuj_Callback(hObject,
lamahuj=str2double(get(hObject,'string'));
handles.lamahuj=lamahuj;
guidata(hObject, handles);
```

5. Dengan langkah yang sama, masukan juga sourcecode dibawah ini untuk edittext3

```
function debit_Callback(hObject, eventdata, handles)
    debit=str2double(get(hObject,'string'));
    handles.debit=debit;
    guidata(hObject, handles);
```

6. Lakukan langkah yang sama untuk tombol running, yang akan digunakan sebagai tombol untuk melakukan proses perhitungan sistem fuzzy dengan mengambil nilai input dari komponen edittext yang telah kita masukan kode sebelumnya, kemudian output dari sistem akan di tampilkan pada komponen edittext hasil prediksi

```
function run_Callback(hObject, eventdata, handles)
    a=readfis('prediksi')
    out=evalfis([handles.Chujan handles.lamahuj handles.debit], a)
    set(handles.prediksi, 'string', out);
```

7. Masukan juga code untuk tombol keluar yang berfungsi untuk menutup form aplikasi

```
function exit_Callback(hObject, eventdata, handles)
    % hObject    handle to exit (see GCBO)
    % eventdata  reserved - to be defined in a future version of MATLAB
    % handles    structure with handles and user data (see GUIDATA)
    close
```

Langkah terakhir adalah melakukan proses testing terhadap aplikasi yang telah kita buat, apakah sudah berfungsi sesuai dengan tujuan dan desain dari rancangan yang telah kita buat.



**Gambar 2.32** Tampilan Aplikasi Deteksi Banjir

Aplikasi ini masih bersifat sangat sederhana sekali, karena hanya bertujuan untuk mengintegrasikan antara model sistem fuzzy yang telah kita buat dengan interface aplikasi yang berbasis GUI sehingga dapat lebih interaktif.



# BAB 3 | MySQL

## A. Pengenalan MySQL

MySQL adalah pengembangan lanjutan dari proyek UNIREG yang dikerjakan oleh Michael Monty Widenius dan TcX (perusahaan perangkat lunak asal Swedia). tetapi, UNIREG belum terlalu kompatibel dengan database dinamis yang dipakai di website. TcX kemudian mencari alternatif lain dan menemukan perangkat lunak yang dikembangkan oleh David Hughes, yaitu miniSQL atau mSQL. Namun, ditemukan masalah lagi karena mSQL tidak mendukung indexing sehingga belum sesuai dengan kebutuhan TcX. Pada akhirnya muncul kerjasama antara pengembang UNIREG (Michael Monty Widenius), mSQL (David Hughes), dan TcX. Kerjasama ini bertujuan untuk mengembangkan sistem database yang baru, dan pada 1995 dirilis MySQL seperti yang dikenal saat ini. Saat ini pengembangan MySQL berada di bawah Oracle.

MySQL adalah sebuah database management system (DBMS) yang menggunakan perintah dasar SQL (Structured Query Language) yang cukup terkenal dan banyak digunakan di seluruh dunia.



**Gambar 3.1** Logo Mysql

MySQL adalah DBMS yang open source dengan dua bentuk lisensi, yaitu Free Software (perangkat lunak bebas) dan Shareware (perangkat lunak berpemilik yang penggunaannya terbatas). Jadi MySQL adalah database server yang gratis dengan lisensi GNU General Public License (GPL) sehingga dapat Kita pakai untuk keperluan pribadi atau komersil tanpa harus membayar lisensi yang ada.

Sebagai salah satu bentuk aplikasi RDBMS (Relational Database Management System), MySQL memiliki berbagai macam fitur yang sangat penting untuk mendukung fungsi dan kegunaanya, yaitu:

1. Scalability

MySQL saat ini dapat menangani database yang cukup besar. Beberapa organisasi atau perusahaan yang telah menerapkannya antara lain: Yahoo!, Cox Communications, Google, Cisco, Texas Instruments, UPS, Sabre Holdings, HP, The Associated Press, dan lain-lain. Bahkan NASA dan Biro Sensus US telah mengimplementasikan MySQL Solutions. Menurut dokumentasi produk MySQL, beberapa database yang digunakan oleh MySQL AB, perusahaan yang membuat MySQL, berisi 50 juta rekod, dan beberapa pengguna MySQL melaporkan bahwa database mereka berisi 60.000 tabel dan 5 milyar baris.

## 2. Portability

Beberapa sistem operasi yang dapat menjalankan MySQL: Unix, Linux, Windows, OS/2, Solaris, dan MacOS. MySQL juga dapat berjalan pada arsitektur yang berbeda-beda, mulai dari PC pada level bawah sampai level tinggi seperti mainframe.

## 3. Connectivity

MySQL mendukung jaringan secara penuh dan socket TCP/IP, socket Unix, dan named pipes (penamaan). Di lain hal, MySQL dapat diakses dari manapun pada internet, dan multiple (banyak) pengguna dapat mengakses database MySQL secara simultan (bersamaan). MySQL juga menyediakan API (Application Programming Interface) untuk mendukung konektivitas dengan beberapa aplikasi yang ditulis dengan menggunakan C, C++, Perl, PHP, Java, dan Python.

## 4. Security

MySQL meliputi sistem yang kita l untuk mengontrol akses ke data. System menggunakan sebuah host dan struktur berbasis client yang mengontrol siapa saja yang dapat mengakses informasi tertentu dan tingkatan akses ke informasi tersebut. MySQL juga mendukung protokol SSL (Secure Sockets Layer) untuk membuat suatu koneksi yang terenkripsi.

## 5. Speed

Jumlah waktu yang diperlukan sebuah database MySQL untuk merespon request data sama cepatnya bahkan lebih cepat ketimbang RDBMS komersil lainnya. Web site MySQL ([www.mysql.com](http://www.mysql.com)) menyediakan hasil-hasil tes benchmark yang menunjukkan hasil kecepatan dalam penerapan MySQL.

## 6. Ease of Use

MySQL mudah untuk di-instal dan diterapkan. Pengguna dapat mendapatkannya dan menjalankannya dalam beberapa menit setelah men-downloadnya. Pada tingkat administratif, MySQL secara relatif mudah dioptimalkan,

terutama jika dibandingkan dengan produk RDBMS lainnya.

#### 7. Open Source Code

MySQL AB menyediakan source code MySQL bagi siapa saja untuk men-download dan menggunakannya. Filosofi dari open source adalah mengizinkan audiens umum untuk berpartisipasi dalam me-review, menguji coba, dan mengembangkan kode.

SQL sendiri merupakan suatu bahasa yang dipakai di dalam pengambilan data pada relational database atau database yang terstruktur. Jadi MySQL adalah database management system yang menggunakan bahasa SQL sebagai bahasa penghubung antara perangkat lunak aplikasi dengan database server.

### B. Kelebihan MySQL

Meskipun menjadi database yang cukup populer, MySQL tentu mempunyai beberapa kelebihan dan kekurangan dibandingkan dengan database server lainnya. Salah satu kekurangan MySQL adalah performanya turun di saat beberapa database manajemen sistem mampu bekerja baik pada pengelolaan database yang besar.

MySQL mempunyai beberapa kelebihan yang bisa Kita manfaatkan untuk mengembangkan perangkat lunak, seperti:

#### 1. Mendukung Integrasi Dengan Bahasa Pemrograman Lain.

Website atau perangkat lunak terkadang dikembangkan dengan menggunakan berbagai macam bahasa pemrograman, jadi Kita tidak perlu khawatir jika menggunakan MySQL. Maka dari itu, MySQL bisa membantu Kita untuk mengembangkan perangkat lunak yang lebih efektif dan tentu saja lebih mudah dengan integrasi antara bahasa pemrograman.

#### 2. Tidak Membutuhkan RAM Besar.

MySQL dapat dipasang pada server dengan spesifikasi kecil. Jadi tidak perlu khawatir jika Kita hanya mempunyai server

dengan kapasitas 1 GB karena Kita masih bisa menggunakan MySQL sebagai database Kita .

3. Mendukung Multi User.

MySQL dapat dipakai oleh beberapa user dalam waktu bersamaan tanpa membuatnya crash atau berhenti bekerja. Ini dapat Kita manfaatkan ketika mengerjakan proyek yang sifatnya tim sehingga seluruh tim dapat bekerja dalam waktu bersamaan tanpa harus menunggu user lain selesai.

4. Bersifat Open Source

MySQL adalah sistem manajemen database gratis. Meskipun gratis, bukan berarti database ini mempunyai kinerja buruk. Apalagi lisensi gratis yang dipakai adalah GPL di bawah pengelolaan Oracle sehingga kualitasnya termasuk baik. Selain itu, Kita juga tidak perlu khawatir jika terjadi masalah karena banyak komunitas dan dokumentasi yang membahas soal MySQL.

5. Struktur Tabel yang Fleksibel.

MySQL mempunyai struktur tabel yang mudah dipakai dan fleksibel. Contohnya saat MySQL memproses ALTER TABLE dan lain sebagainya. Jika dibandingkan dengan database lain seperti Oracle dan PostgreSQL, MySQL tergolong lebih mudah.

6. Tipe Data yang Bervariasi.

Kelebihan lain dari MySQL adalah mendukung berbagai macam data yang bisa Kita gunakan di MySQL. Contohnya float, integer, date, char, text, timestamp, double, dan lain sebagainya. Jadi manajemen database sistem ini sangat membantu Kita untuk mengembangkan perangkat lunak yang berguna untuk pengelolaan database di server.

7. Keamanan yang Terjamin.

Open source bukan berarti MySQL menyediakan keamanan yang buruk. Malah sebaliknya, MySQL mempunyai fitur keamanan yang cukup apik. Ada beberapa lapisan keamanan yang diterapkan oleh MySQL, seperti level nama host, dan

subnetmask. Selain itu MySQL juga dapat mengatur hak akses user dengan enkripsi password tingkat tinggi.

### **C. Kekurangan MySQL**

Walaupun Mysql memiliki banyak kelebihan, tetapi juga memiliki beberapa kelemahan yang dimiliki oleh MySQL sehingga Kita perlu mempertimbangkannya juga sebelum memakainya.

#### **1. Kurang Cocok untuk Aplikasi Game dan Mobile**

Kebanyakan pengembang game maupun aplikasi mobile tidak menggunakannya karena memang database manajemen sistem ini masih kurang bagus dipakai untuk sistem aplikasi tersebut.

#### **2. Sulit Mengelola Database yang Besar**

Jika Kita ingin mengembangkan aplikasi atau sistem di perusahaan dengan database yang cukup besar, ada baiknya jika menggunakan database manajemen sistem selain MySQL. MySQL dikembangkan supaya ramah dengan perangkat yang mempunyai spesifikasi rendah, itulah mengapa MySQL tidak memiliki fitur yang lengkap seperti aplikasi lainnya

#### **3. Technical Support yang Kurang Bagus**

Sifatnya yang open source terkadang membuat aplikasi tidak menyediakan technical support yang memadai. Technical support MySQL diklaim kurang bagus. Hal ini membuat pengguna kesulitan. Apalagi jika pengguna mengalami masalah yang berhubungan dengan pengoperasian perangkat lunak tersebut dan membutuhkan bantuan technical support.

### **D. Structure Query Language (SQL)**

Sesuai dengan pengertian MySQL yang sudah dibahas sebelumnya, bahwa sistem manajemen server ini menggunakan SQL sebagai bahasa pemrogramannya. SQL merupakan bahasa pemrograman yang perlu Kita pahami karena dapat merelasikan antara beberapa tabel dengan database maupun antar database. Ada tiga bentuk SQL yang perlu Kita ketahui,

yaitu Data Definition Language (DDL), Data Manipulation Language (DML), dan Data Control Language (DCL).

#### 1. Data Definition Language (DDL)

DDL berguna pada saat Kita ingin mendefinisikan data di dalam database. Terdapat beberapa query yang dikelompokkan ke dalam DDL, yaitu:

- a. CREATE : Dipakai untuk membuat tabel dan database.
- b. DROP : Dipakai untuk menghapus database dan tabel.
- c. ALTER : Dipakai untuk mengubah struktur tabel yang sudah ada.

Alter dapat mengganti field menggunakan perintah "Change", menambahkan field menggunakan perintah "Add", atau menghapus field menggunakan perintah "drop", dan mengubah namanya menggunakan perintah "Rename".

Argumen DDL di atas perlu Kita pahami karena merupakan dasar penggunaan SQL di bagian awal pembuatan database. Contohnya saja jika belum menjalankan perintah "CREATE", Kita belum bisa melanjutkan penggunaan argumen yang lainnya.

#### 2. Data Manipulation Language (DML)

DML dapat Kita pakai setelah menjalankan perintah DDL. DML berfungsi untuk memanipulasi, mengubah, atau mengganti isi dari database (tabel) yang sudah ada. Terdapat beberapa perintah DML yang perlu Kita ketahui, yaitu:

- a. INSERT berfungsi untuk memasukkan data ke dalam tabel pada database.
- b. UPDATE berfungsi untuk mengubah data yang ada di dalam tabel pada database.
- c. DELETE berfungsi untuk menghapus data di dalam tabel pada database.

#### 3. Data Control Language (DCL)

Jika Kita sudah mempunyai user dan ingin mengatur hak akses masing-masing user, Kita sebaiknya memahami berbagai macam jenis DCL dan cara penggunaannya. DCL berguna untuk memberikan hak akses database,

mendefinisikan space, mengalokasikan space, dan melakukan audit penggunaan database.

Terdapat beberapa perintah DCL yang perlu Kita ketahui, yaitu:

- a. GRANT berfungsi untuk memberikan izin kepada user untuk mengakses database.
- b. REVOKE berfungsi untuk membatalkan izin user untuk mengakses database.
- c. COMMIT berfungsi untuk menetapkan penyimpanan pada database.
- d. ROLLBACK berfungsi untuk membatalkan penyimpanan pada database.



# BAB 4 | STUDI KASUS: MEMBANGUN APLIKASI BERBASIS AI

## SISTEM PENILAIAN KINERJA KARYAWAN MENGUNAKAN FIS METODE MAMDANI

### A. Pengantar

Studi kasus ini akan menerapkan *fuzzy inference system* metode mamdani pada sistem penilaian kinerja karyawan. Dimana masalah yang biasa dihadapi pada sistem penilaian kinerja karyawan secara konvensional adalah pihak manajemen mengalami kesulitan dalam melakukan penilaian kinerja karyawan dan juga proses perhitungan hasil kinerja karyawan yang relatif lama. Studi kasus ini dibuat berdasarkan rujukan beberapa penelitian dengan objek penelitian yang sama, antara lain:

1. Penelitian yang dilakukan oleh G.Vasanti pada tahun 2017 dengan judul "*Teacher's Performance Appraisal System Using Fuzzy Logic- A Case Study*". Pada penelitian ini metode yang digunakan adalah *fuzzy logic* atau logika *fuzzy*, dalam penelitian ini. Membahas tentang penilaian kualitatif untuk mengevaluasi kinerja, dimana proses penilaian dari atasan langsung, kemudian penilaian linguistik tersebut diubah menjadi bilangan *fuzzy* dan akan dihitung skor nilainya pada sistem penilaian kinerja menggunakan logika *fuzzy*. Penelitian mengajukan logika *fuzzy* pada sistem penilaian kinerja sebagai bentuk modifikasi dan sistem yang lebih layak digunakan di organisasi atau institusi pendidikan

tersebut. Hasil eksperimen dan pengujian yang dilakukan menyatakan bahwa logika *fuzzy* layak diterapkan pada sistem penilaian kinerja guru atau pengajar karena dapat menyajikan informasi yang lengkap, akurat dan tidak bias serta meningkatkan keaslian penilaian.

2. Penelitian yang dilakukan oleh Mutmainah, Umi Marfuah dan Andreas Tri Panudju pada tahun 2017 dengan judul "*Employee Performance Appraisal Model Using Human Resources Scorecard And Analytical Hierarchy Process (AHP)*". Pada penelitian ini metode yang digunakan adalah *Human Resources Scorecard* dan *Analytical Hierarchy Process (AHP)*, dalam penelitian ini. Membahas tentang penilaian kualitatif untuk mengevaluasi kinerja, dimana proses penilaian dari atasan langsung, kemudian penilaian linguistik tersebut diubah menjadi bilangan *fuzzy* dan akan dihitung skor nilainya pada sistem penilaian kinerja menggunakan logika *fuzzy*. Penelitian mengajukan logika *fuzzy* pada sistem penilaian kinerja sebagai bentuk modifikasi dan sistem yang lebih layak digunakan di organisasi atau institusi pendidikan tersebut. Hasil eksperimen dan pengujian yang dilakukan menyatakan bahwa logika *fuzzy* layak diterapkan pada sistem penilaian kinerja guru atau pengajar karena dapat menyajikan informasi yang lengkap, akurat dan tidak bias serta meningkatkan keaslian penilaian.
3. Penelitian yang dilakukan oleh Alamsyah dan Izza Hasanul Muna pada tahun 2016 dengan judul "*Metode Fuzzy Inference System untuk Penilaian Kinerja Pegawai Perpustakaan dan Pustakawan*". Pada penelitian ini metode yang digunakan adalah *Fuzzy Inference System* metode sugeno. Dimana masalah yang terjadi adalah sistem penilaian kinerja yang telah dilakukan, kebanyakan penekanannya masih tertuju pada aspek kedisiplinan, sementara kinerja pegawai tidak cukup hanya dilihat dari aspek kedisiplinan saja, tetapi juga aspek-aspek yang lainnya. Hasil pengujian dan analisis yang telah dilakukan terhadap 50 pegawai, diperoleh skor kinerja

pegawai tertinggi yaitu 90,89 dan skor kinerja terendah adalah sebesar 80,77.

4. Penelitian yang dilakukan oleh Alamsyah dan Izza Hasanul Muna pada tahun 2016 dengan judul penelitian “Metode *fuzzy inference system* untuk Penilaian Kinerja Pegawai Perpustakaan dan Pustakawan”. Dimana metode yang digunakan Pada penelitian ini adalah *fuzzy inference system* metode sugeno, sedangkan masalah yang diangkat dalam penelitian adalah sistem yang selama ini penilaian kinerja yang telah dilakukan, kebanyakan penekanannya masih tertuju pada aspek kedisiplinan, sementara kinerja pegawai tidak cukup hanya dilihat dari aspek kedisiplinan saja, tetapi juga aspek-aspek yang lainnya. Setelah melakukan beberapa pengujian terhadap metode FIS untuk mengukur kinerja pegawai perpustakaan dan pustakawan, dapat ditarik simpulan yaitu FIS diterapkan pada tahap kalkulasi penilaian kinerja pegawai, dimana nilai dari tiap-tiap variabel yang telah diinputkan dilakukan fuzzifikasi terlebih dahulu. Selanjutnya dilakukan inferensi terhadap aturan yang dipakai dan diakhiri dengan tahap defuzzifikasi yang berupa penghitungan skor menggunakan metode berbobot rata-rata. Dengan metode ini diperoleh skor kinerja tertinggi yaitu 90,89 dan skor kinerja terendah sebesar 80,77.
5. Penelitian yang dilakukan oleh Yu-Wei Chang pada tahun 2015 dengan judul “*Employee Performance Appraisal in a Logistics Company*”. Dalam penelitian ini menggunakan metode *Analytic Hierarchy Process* (AHP) dan metode *Technique For Order Preference By Similarity To An Ideal Solution* (TOPSIS) untuk melakukan penilaian kinerja pegawai. Pada penelitian ini mencoba menggabungkan metode AHP dengan metode TOPSIS, dimana metode AHP digunakan untuk proses pemberian bobot pada kriteria sedangkan metode TOPSIS digunakan untuk proses perbandingan atau peringkat kinerja keseluruhan karyawan. Hasil menunjukkan bahwa metode yang diusulkan adalah metode umum dan berlaku untuk masalah penilaian kinerja. Hasil dari

penelitian ini menyatakan bahwa model dapat digunakan untuk mengpenilaian kinerja karyawan secara efektif

6. Penelitian yang dilakukan oleh Ades Galih Anto, Hindayati Mustafidah dan Aman Suyadi pada tahun 2015 dengan judul “Sistem Pendukung Keputusan Penilaian Kinerja Karyawan Menggunakan Metode SAW”. metode yang digunakan pada penelitian ini adalah metode *simple additive weighting* (SAW), sedangkan masalah yang dihadapi adalah kondisi Lembaga Penjaminan Mutu Universitas Muhammadiyah Purwokerto (LPM) belum mempunyai aplikasi khusus yang digunakan untuk penilaian kinerja, dimana proses penilaian secara manual sangat mungkin terjadi kesalahan dalam menghitung setiap kriteria, serta memakan waktu lama dalam proses perhitungannya. Adapun hasil penelitian ini menyimpulkan bahwa aplikasi sistem pendukung keputusan penilaian kinerja karyawan berhasil dibangun sesuai dengan rancangan atau desainnya dan aplikasi sistem pendukung keputusan penilaian kinerja karyawan dapat digunakan oleh lembaga penjaminan mutu universitas muhammadiyah purwokerto untuk menilai kinerja karyawan secara obyektif.
7. Penelitian yang dilakukan oleh Okoye Ubaka M.N. , Adewole A.P. , Folorunso O. dan Ezike J.O.J. pada tahun 2013 dengan judul “*Neural Network Model for Performance Evaluation of Academic Staff of Tertiary Institutions*”. Pada penelitian ini metode yang digunakan adalah *neural network* atau jaringan saraf tiruan, dalam penelitian ini mencoba membandingkan antara model *parametrics* dengan *neural network* atau jaringan saraf tiruan dengan tujuan mencari tahu metode yang lebih baik dalam penilaian kinerja staff akademik. berdasarkan pengujian yang telah dilakukan dinyatakan bahwa hasil penelitian menunjukkan bahwa *Multilayer Perceptron Neural Network* (MPNN) dengan algoritma *backpropagation* mendapatkan hasil yang lebih baik jika dibandingkan dengan model *parametrics*, dengan RSME minimum sebesar 90%.

## **B. Karyawan**

Karyawan adalah orang pribadi yang bekerja pada pemberi kerja, berdasarkan perjanjian atau kesepakatan kerja baik secara tertulis maupun tidak tertulis, untuk melaksanakan suatu pekerjaan dalam jabatan atau kegiatan tertentu dengan memperoleh imbalan yang dibayarkan berdasarkan periode tertentu, penyelesaian pekerjaan, atau ketentuan lain yang ditetapkan pemberi kerja, termasuk orang pribadi yang melakukan pekerjaan dalam jabatan negeri. Secara umum karyawan di bedakan menjadi dua, yaitu:

1. Karyawan tetap adalah aset utama perusahaan yang menjadi perencana dan pelaku aktif dari aktivitas organisasi (Berutu, 2015), peran karyawan bagi suatu perusahaan sangat penting dalam keberhasilan suatu perusahaan, karena dengan tenaga mereka suatu perusahaan bisa berjalan dan perusahaan akan menghasilkan kualitas yang baik.
2. Karyawan kontrak adalah karyawan yang diperbantukan untuk menyelesaikan pekerjaan-pekerjaan rutin perusahaan dan tidak ada jaminan kelangsungan masa kerjanya.

Dari uraian diatas, dapat dikatakan bahwa status sebagai karyawan tetap merupakan salah satu bentuk peningkatan karir bagi karyawan kontrak. Karyawan tetap adalah aset utama perusahaan yang menjadi perencana dan pelaku aktif dari aktivitas organisasi

## **C. Kinerja Karyawan**

Karyawan merupakan salah satu elemen penting dalam sebuah perusahaan, karena keberhasilan perusahaan ditentukan oleh kinerja karyawannya (Sugijono, 2015), Penilaian kinerja (*performance appraisal*) adalah proses mengevaluasi seberapa baik karyawan melakukan pekerjaan mereka jika dibandingkan dengan seperangkat stkitar, dan kemudian mengkomunikasikan informasi tersebut kepada karyawan (Setiyowati, 2013). Penilaian kinerja juga disebut pemeringkatan karyawan, evaluasi karyawan, tinjauan kerja (Anto, Mustafidah, & Suyadi, 2015), evaluasi kinerja, dan penilaian hasil. Secara

umum dapat didefenisikan bahwa penilaian kinerja tersebut adalah sebagai penilaian hasil kerja nyata dengan stkitar kualitas maupun kuantitas yang dihasilkan oleh setiap pegawai. Penilaian kinerja pegawai mutlak harus dilakukan untuk mengetahui prestasi yang dapat dicapai setiap pegawai.

Apakah prestasi yang dicapai setiap pegawai baik, sedang, kurang. Penilaian prestasi penting bagi setiap pegawai dan berguna bagi organisasi untuk mengambil keputusan dan menetapkan tindakan kebijaksanaan selanjutnya (Haryadi & Rahman, 2017), bagi perusahaan manfaat ataupun keuntungan dari pelaksanaan penilaian kinerja adalah untuk perbaikan seluruh simpul unit-unit yang ada dalam perusahaan, untuk memotivasi karyawan dan mengembangkan ketrampilan karyawannya serta untuk meningkatkan kualitas komunikasi dan meningkatkan motivasi karyawan (Sugijono, 2015). Penilaian kinerja (*performance appraisal*) adalah proses mengevaluasi seberapa baik karyawan melakukan pekerjaan mereka jika dibandingkan dengan seperangkat standar, dan kemudian mengkomunikasikan informasi tersebut kepada karyawan (Setiyowati, 2013). Penilaian kinerja juga disebut pemeringkatan karyawan, evaluasi karyawan, tinjauan kerja (Anto, Mustafidah, & Suyadi, 2015), evaluasi kinerja, dan penilaian hasil. Secara umum dapat didefenisikan bahwa penilaian kinerja tersebut adalah sebagai penilaian hasil kerja nyata dengan standar kualitas maupun kuantitas yang dihasilkan oleh setiap pegawai. Penilaian kinerja pegawai mutlak harus dilakukan untuk mengetahui prestasi yang dapat dicapai setiap pegawai.

Apakah prestasi yang dicapai setiap pegawai baik, sedang, kurang. Penilaian prestasi penting bagi setiap pegawai dan berguna bagi organisasi untuk mengambil keputusan dan menetapkan tindakan kebijaksanaan selanjutnya (Haryadi & Rahman, 2017), bagi perusahaan manfaat ataupun keuntungan dari pelaksanaan penilaian kinerja adalah untuk perbaikan seluruh simpul unit-unit yang ada dalam perusahaan, untuk memotivasi karyawan

dan mengembangkan ketrampilan karyawannya serta untuk meningkatkan kualitas komunikasi dan meningkatkan motivasi karyawan (Sugijono, 2015).

#### **D. Komponen Penilaian Kinerja Karyawan**

Kriteria adalah ukuran, aturan atau standar-standar yang memandu suatu pengambilan keputusan (Anto, Mustafidah, & Suyadi, 2015). Kriteria yang dimaksud dalam penelitian ini adalah kriteria yang digunakan perusahaan, yaitu:

1. Absensi

Kriteria absensi dinilai berdasarkan dari *persentase* kehadiran karyawan pada hari kerja.

2. Prestasi

Prestasi Kerja adalah hasil kerja yang dicapai oleh seorang pegawai dalam melaksanakan tugas yang dibebankan kepadanya. Pada umumnya, prestasi kerja seorang pegawai dipengaruhi oleh kecakapan, ketrampilan, pengalaman, dan kesungguhan pegawai yang bersangkutan dalam melaksanakan pekerjaannya. Penilai menilai hasil kerja baik kualitas maupun kuantitas yang dapat dihasilkan pegawai tersebut dari uraian pekerjaannya

3. Disiplin

Ketaatan adalah kesanggupan seorang pegawai untuk menaati segala peraturan yang berlaku, menaati perintah kedinasan yang diberikan oleh atasan yang berwenang, serta kesanggupan untuk tidak melanggar larangan yang ditentukan

4. Loyalitas

Karyawan dituntut harus loyal atau setia pada perusahaan, selalu berusaha menjunjung tinggi nama baik dan juga keberlangsungan kegiatan perusahaan.

#### **E. Desain dan Rancangan Sistem Fuzzy**

Seperti yang telah dijelaskan diatas terkait dengan komponen penilaian kinerja karyawan, maka komponen-komponen tersebut dijadikan sebagai *input* untuk sistem

penilaian kinerja yang selanjutnya disebut dengan variabel untuk masing-masing dengan perincian sebagai berikut:

1. Variabel absensi dengan semesta pembicaraan mulai dari 0 sampai dengan 100 dengan satuan persen.
2. Variabel prestasi dengan semesta pembicaraan mulai dari 0 sampai dengan 100 dengan satuan poin.
3. Variabel disiplin dengan semesta pembicaraan mulai dari 0 sampai dengan 100 dengan satuan poin.
4. Variabel Loyalitas dengan semesta pembicaraan mulai dari 0 sampai dengan 100 dengan satuan poin.

## F. Desain dan Rancangan Fungsi Keanggotaan

Setelah variabel ditentukan selanjutnya variabel-variabel tersebut dibagi lagi menjadi beberapa himpunan seperti dibawah ini.

### 1. Variabel *Input*

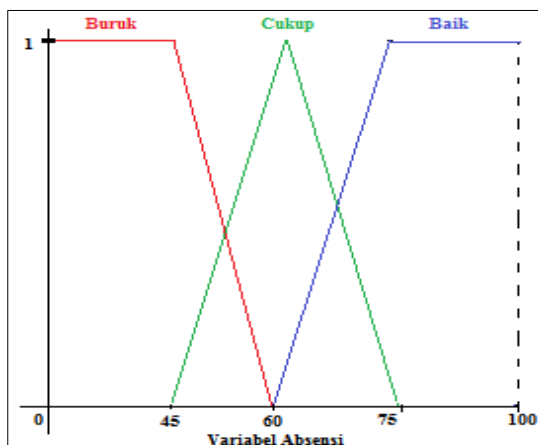
#### a. Variabel Absensi (0-100%)

Rendah : [0-60]

Cukup : [45-75]

Tinggi: [60-100]

Dengan fungsi keanggotaan seperti pada gambar dibawah ini.



**Gambar 4.1** Fungsi Keanggotaan Variabel Absensi



b. Variabel Prestasi (0-100 poin)

Sangat Rendah : [0-25]

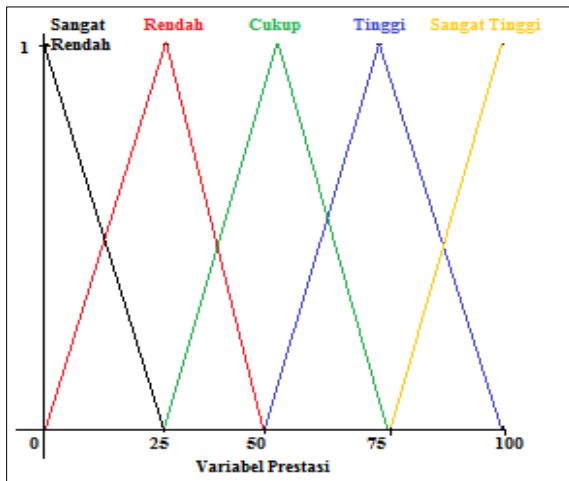
Rendah : [25-50]

Cukup : [25-75]

Tinggi : [50-100]

Sangat Tinggi : [75-100]

Dengan fungsi keanggotaan seperti pada gambar dibawah ini.



**Gambar 4.2** Fungsi Keanggotaan Variabel Prestasi

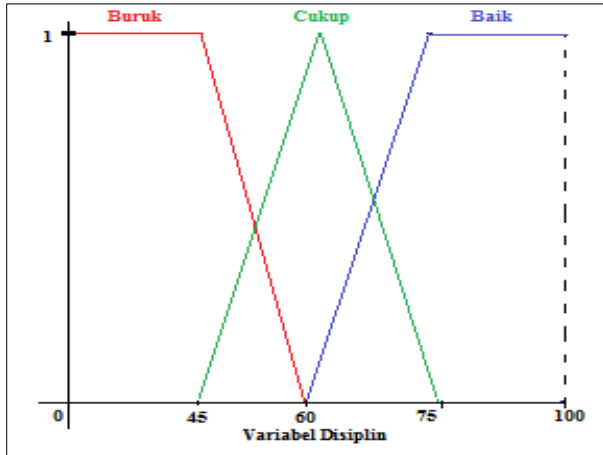
c. Variabel Disiplin

Buruk : [0-60]

Cukup : [45-75]

Baik : [75-100]

Dengan fungsi keanggotaan seperti pada gambar dibawah ini.

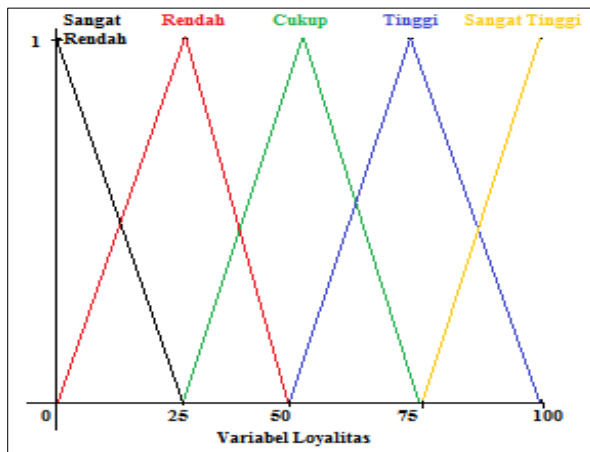


**Gambar 4.3** Fungsi Keanggotaan Variabel Disiplin

d. Variabel Loyalitas

Sangat Rendah : [0-25]  
 Rendah : [0-50]  
 Cukup : [25-75]  
 Tinggi : [50-100]  
 Sangat Tinggi : [75-100]

Dengan fungsi keanggotaan seperti pada gambar dibawah ini.



**Gambar 4.4** Fungsi Keanggotaan Variabel Loyalitas

2. Variabel *output* yang ditentukan terbagi menjadi:

a. Variabel Status Kinerja

Sangat Rendah : [0-50]

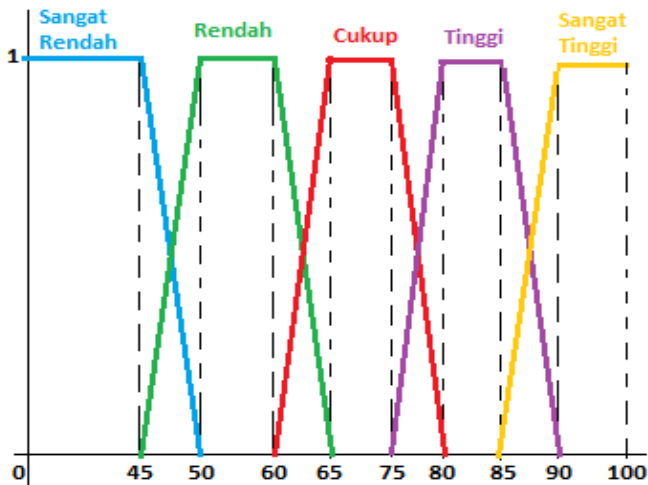
Rendah : [45-65]

Cukup : [60-80]

Tinggi : [75-90]

Sangat Tinggi : [85-100]

Seperti yang ditunjukkan pada gambar dibawah ini



**Gambar 4.5** Fungsi Keanggotaan Variabel Status Kinerja

### G. Desain dan Rancangan Rule System Fuzzy

Pada tahap ini kita harus merancang basis aturan dasar dari sistem fuzzy, dimana tahap pemberian aturan atau inferensi ini merupakan proses pemberian aturan sebagai basis pengetahuan seorang ahli. Adapun rancangan rule base sistem fuzzy dapat dilihat pada tabel berikut:

**Tabel 4.1** Rancangan Rule Base

No	Variabel Input				Variabel Output
	Absensi	Prestasi	Disiplin	Loyalitas	Kinerja
1	Buruk	Sangat Rendah	Buruk	Sangat Rendah	Sangat Rendah
2	Buruk	Sangat Rendah	Buruk	Rendah	Sangat Rendah
3	Buruk	Sangat Rendah	Buruk	Cukup	Sangat Rendah
4	Buruk	Sangat Rendah	Buruk	Tinggi	Rendah
5	Buruk	Sangat Rendah	Buruk	Sangat Tinggi	Rendah
6	Buruk	Sangat Rendah	Cukup	Sangat Rendah	Sangat Rendah
7	Buruk	Sangat Rendah	Cukup	Rendah	Sangat Rendah
8	Buruk	Sangat Rendah	Cukup	Cukup	Sangat Rendah
9	Buruk	Sangat Rendah	Cukup	Tinggi	Rendah
10	Buruk	Sangat Rendah	Cukup	Sangat Tinggi	Rendah
11	Buruk	Sangat Rendah	Baik	Sangat Rendah	Sangat Rendah
12	Buruk	Sangat Rendah	Baik	Rendah	Sangat Rendah
13	Buruk	Sangat Rendah	Baik	Cukup	Rendah
14	Buruk	Sangat Rendah	Baik	Tinggi	Rendah
15	Buruk	Sangat Rendah	Baik	Sangat Tinggi	Cukup

No	Variabel Input				Variabel Output
	Absensi	Prestasi	Disiplin	Loyalitas	Kinerja
16	Buruk	Rendah	Buruk	Sangat Rendah	Sangat Rendah
17	Buruk	Rendah	Buruk	Rendah	Sangat Rendah
18	Buruk	Rendah	Buruk	Cukup	Sangat Rendah
19	Buruk	Rendah	Buruk	Tinggi	Rendah
20	Buruk	Rendah	Buruk	Sangat Tinggi	Rendah
21	Buruk	Rendah	Cukup	Sangat Rendah	Sangat Rendah
22	Buruk	Rendah	Cukup	Rendah	Sangat Rendah
23	Buruk	Rendah	Cukup	Cukup	Rendah
24	Buruk	Rendah	Cukup	Tinggi	Cukup
25	Buruk	Rendah	Cukup	Sangat Tinggi	Cukup
26	Buruk	Rendah	Baik	Sangat Rendah	Sangat Rendah
27	Buruk	Rendah	Baik	Rendah	Sangat Rendah
28	Buruk	Rendah	Baik	Cukup	Rendah
29	Buruk	Rendah	Baik	Tinggi	Cukup
30	Buruk	Rendah	Baik	Sangat Tinggi	Cukup
31	Buruk	Cukup	Buruk	Sangat Rendah	Sangat Rendah
32	Buruk	Cukup	Buruk	Rendah	Sangat Rendah

No	Variabel Input				Variabel Output
	Absensi	Prestasi	Disiplin	Loyalitas	Kinerja
33	Buruk	Cukup	Buruk	Cukup	Rendah
34	Buruk	Cukup	Buruk	Tinggi	Cukup
35	Buruk	Cukup	Buruk	Sangat Tinggi	Cukup
36	Buruk	Cukup	Cukup	Sangat Rendah	Sangat Rendah
37	Buruk	Cukup	Cukup	Rendah	Rendah
38	Buruk	Cukup	Cukup	Cukup	Rendah
39	Buruk	Cukup	Cukup	Tinggi	Cukup
40	Buruk	Cukup	Cukup	Sangat Tinggi	Cukup
41	Buruk	Cukup	Baik	Sangat Rendah	Rendah
42	Buruk	Cukup	Baik	Rendah	Rendah
43	Buruk	Cukup	Baik	Cukup	Cukup
44	Buruk	Cukup	Baik	Tinggi	Cukup
45	Buruk	Cukup	Baik	Sangat Tinggi	Tinggi
46	Buruk	Tinggi	Buruk	Sangat Rendah	Rendah
47	Buruk	Tinggi	Buruk	Rendah	Rendah
48	Buruk	Tinggi	Buruk	Cukup	Cukup
49	Buruk	Tinggi	Buruk	Tinggi	Cukup

No	Variabel Input				Variabel Output
	Absensi	Prestasi	Disiplin	Loyalitas	Kinerja
50	Buruk	Tinggi	Buruk	Sangat Tinggi	Cukup
51	Buruk	Tinggi	Cukup	Sangat Rendah	Rendah
52	Buruk	Tinggi	Cukup	Rendah	Rendah
53	Buruk	Tinggi	Cukup	Cukup	Cukup
54	Buruk	Tinggi	Cukup	Tinggi	Cukup
55	Buruk	Tinggi	Cukup	Sangat Tinggi	Tinggi
56	Buruk	Tinggi	Baik	Sangat Rendah	Cukup
57	Buruk	Tinggi	Baik	Rendah	Cukup
58	Buruk	Tinggi	Baik	Cukup	Tinggi
59	Buruk	Tinggi	Baik	Tinggi	Tinggi
60	Buruk	Tinggi	Baik	Sangat Tinggi	Tinggi
61	Buruk	Sangat Tinggi	Buruk	Sangat Rendah	Rendah
62	Buruk	Sangat Tinggi	Buruk	Rendah	Rendah
63	Buruk	Sangat Tinggi	Buruk	Cukup	Rendah
64	Buruk	Sangat Tinggi	Buruk	Tinggi	Cukup
65	Buruk	Sangat Tinggi	Buruk	Sangat Tinggi	Tinggi
66	Buruk	Sangat Tinggi	Cukup	Sangat Rendah	Cukup

No	Variabel Input				Variabel Output
	Absensi	Prestasi	Disiplin	Loyalitas	Kinerja
67	Buruk	Sangat Tinggi	Cukup	Rendah	Cukup
68	Buruk	Sangat Tinggi	Cukup	Cukup	Cukup
69	Buruk	Sangat Tinggi	Cukup	Tinggi	Tinggi
70	Buruk	Sangat Tinggi	Cukup	Sangat Tinggi	Tinggi
71	Buruk	Sangat Tinggi	Baik	Sangat Rendah	Cukup
72	Buruk	Sangat Tinggi	Baik	Rendah	Cukup
73	Buruk	Sangat Tinggi	Baik	Cukup	Tinggi
74	Buruk	Sangat Tinggi	Baik	Tinggi	Tinggi
75	Buruk	Sangat Tinggi	Baik	Sangat Tinggi	Tinggi
76	Cukup	Sangat Rendah	Buruk	Sangat Rendah	Sangat Rendah
77	Cukup	Sangat Rendah	Buruk	Rendah	Sangat Rendah
78	Cukup	Sangat Rendah	Buruk	Cukup	Sangat Rendah
79	Cukup	Sangat Rendah	Buruk	Tinggi	Rendah
80	Cukup	Sangat Rendah	Buruk	Sangat Tinggi	Rendah
81	Cukup	Sangat Rendah	Cukup	Sangat Rendah	Sangat Rendah
82	Cukup	Sangat Rendah	Cukup	Rendah	Rendah
83	Cukup	Sangat Rendah	Cukup	Cukup	Rendah



No	Variabel Input				Variabel Output
	Absensi	Prestasi	Disiplin	Loyalitas	Kinerja
84	Cukup	Sangat Rendah	Cukup	Tinggi	Cukup
85	Cukup	Sangat Rendah	Cukup	Sangat Tinggi	Cukup
86	Cukup	Sangat Rendah	Baik	Sangat Rendah	Rendah
87	Cukup	Sangat Rendah	Baik	Rendah	Cukup
88	Cukup	Sangat Rendah	Baik	Cukup	Cukup
89	Cukup	Sangat Rendah	Baik	Tinggi	Tinggi
90	Cukup	Sangat Rendah	Baik	Sangat Tinggi	Tinggi
91	Cukup	Rendah	Buruk	Sangat Rendah	Sangat Rendah
92	Cukup	Rendah	Buruk	Rendah	Sangat Rendah
93	Cukup	Rendah	Buruk	Cukup	Rendah
94	Cukup	Rendah	Buruk	Tinggi	Rendah
95	Cukup	Rendah	Buruk	Sangat Tinggi	Cukup
96	Cukup	Rendah	Cukup	Sangat Rendah	Rendah
97	Cukup	Rendah	Cukup	Rendah	Rendah
98	Cukup	Rendah	Cukup	Cukup	Cukup
99	Cukup	Rendah	Cukup	Tinggi	Cukup
100	Cukup	Rendah	Cukup	Sangat Tinggi	Cukup

No	Variabel Input				Variabel Output
	Absensi	Prestasi	Disiplin	Loyalitas	Kinerja
101	Cukup	Rendah	Baik	Sangat Rendah	Rendah
102	Cukup	Rendah	Baik	Rendah	Rendah
103	Cukup	Rendah	Baik	Cukup	Cukup
104	Cukup	Rendah	Baik	Tinggi	Cukup
105	Cukup	Rendah	Baik	Sangat Tinggi	Tinggi
106	Cukup	Cukup	Buruk	Sangat Rendah	Rendah
107	Cukup	Cukup	Buruk	Rendah	Rendah
108	Cukup	Cukup	Buruk	Cukup	Cukup
109	Cukup	Cukup	Buruk	Tinggi	Cukup
110	Cukup	Cukup	Buruk	Sangat Tinggi	Cukup
111	Cukup	Cukup	Cukup	Sangat Rendah	Rendah
112	Cukup	Cukup	Cukup	Rendah	Cukup
113	Cukup	Cukup	Cukup	Cukup	Cukup
114	Cukup	Cukup	Cukup	Tinggi	Tinggi
115	Cukup	Cukup	Cukup	Sangat Tinggi	Tinggi
116	Cukup	Cukup	Baik	Sangat Rendah	Cukup
117	Cukup	Cukup	Baik	Rendah	Cukup

No	Variabel Input				Variabel Output
	Absensi	Prestasi	Disiplin	Loyalitas	Kinerja
118	Cukup	Cukup	Baik	Cukup	Cukup
119	Cukup	Cukup	Baik	Tinggi	Tinggi
120	Cukup	Cukup	Baik	Sangat Tinggi	Tinggi
121	Cukup	Tinggi	Buruk	Sangat Rendah	Rendah
122	Cukup	Tinggi	Buruk	Rendah	Rendah
123	Cukup	Tinggi	Buruk	Cukup	Cukup
124	Cukup	Tinggi	Buruk	Tinggi	Cukup
125	Cukup	Tinggi	Buruk	Sangat Tinggi	Cukup
126	Cukup	Tinggi	Cukup	Sangat Rendah	Rendah
127	Cukup	Tinggi	Cukup	Rendah	Rendah
128	Cukup	Tinggi	Cukup	Cukup	Cukup
129	Cukup	Tinggi	Cukup	Tinggi	Cukup
130	Cukup	Tinggi	Cukup	Sangat Tinggi	Tinggi
131	Cukup	Tinggi	Baik	Sangat Rendah	Rendah
132	Cukup	Tinggi	Baik	Rendah	Cukup
133	Cukup	Tinggi	Baik	Cukup	Tinggi
134	Cukup	Tinggi	Baik	Tinggi	Sangat Tinggi

No	Variabel Input				Variabel Output
	Absensi	Prestasi	Disiplin	Loyalitas	Kinerja
135	Cukup	Tinggi	Baik	Sangat Tinggi	Sangat Tinggi
136	Cukup	Sangat Tinggi	Buruk	Sangat Rendah	Cukup
137	Cukup	Sangat Tinggi	Buruk	Rendah	Cukup
138	Cukup	Sangat Tinggi	Buruk	Cukup	Tinggi
139	Cukup	Sangat Tinggi	Buruk	Tinggi	Tinggi
140	Cukup	Sangat Tinggi	Buruk	Sangat Tinggi	Sangat Tinggi
141	Cukup	Sangat Tinggi	Cukup	Sangat Rendah	Cukup
142	Cukup	Sangat Tinggi	Cukup	Rendah	Cukup
143	Cukup	Sangat Tinggi	Cukup	Cukup	Cukup
144	Cukup	Sangat Tinggi	Cukup	Tinggi	Tinggi
145	Cukup	Sangat Tinggi	Cukup	Sangat Tinggi	Tinggi
146	Cukup	Sangat Tinggi	Baik	Sangat Rendah	Cukup
147	Cukup	Sangat Tinggi	Baik	Rendah	Cukup
148	Cukup	Sangat Tinggi	Baik	Cukup	Tinggi
149	Cukup	Sangat Tinggi	Baik	Tinggi	Tinggi
150	Cukup	Sangat Tinggi	Baik	Sangat Tinggi	Tinggi
151	Baik	Sangat Rendah	Buruk	Sangat Rendah	Sangat Rendah

No	Variabel Input				Variabel Output
	Absensi	Prestasi	Disiplin	Loyalitas	Kinerja
152	Baik	Sangat Rendah	Buruk	Rendah	Rendah
153	Baik	Sangat Rendah	Buruk	Cukup	Rendah
154	Baik	Sangat Rendah	Buruk	Tinggi	Cukup
155	Baik	Sangat Rendah	Buruk	Sangat Tinggi	Cukup
156	Baik	Sangat Rendah	Cukup	Sangat Rendah	Rendah
157	Baik	Sangat Rendah	Cukup	Rendah	Rendah
158	Baik	Sangat Rendah	Cukup	Cukup	Cukup
159	Baik	Sangat Rendah	Cukup	Tinggi	Cukup
160	Baik	Sangat Rendah	Cukup	Sangat Tinggi	Tinggi
161	Baik	Sangat Rendah	Baik	Sangat Rendah	Rendah
162	Baik	Sangat Rendah	Baik	Rendah	Rendah
163	Baik	Sangat Rendah	Baik	Cukup	Cukup
164	Baik	Sangat Rendah	Baik	Tinggi	Cukup
165	Baik	Sangat Rendah	Baik	Sangat Tinggi	Tinggi
166	Baik	Rendah	Buruk	Sangat Rendah	Sangat Rendah
167	Baik	Rendah	Buruk	Rendah	Rendah
168	Baik	Rendah	Buruk	Cukup	Cukup

No	Variabel Input				Variabel Output
	Absensi	Prestasi	Disiplin	Loyalitas	Kinerja
169	Baik	Rendah	Buruk	Tinggi	Cukup
170	Baik	Rendah	Buruk	Sangat Tinggi	Tinggi
171	Baik	Rendah	Cukup	Sangat Rendah	Rendah
172	Baik	Rendah	Cukup	Rendah	Rendah
173	Baik	Rendah	Cukup	Cukup	Cukup
174	Baik	Rendah	Cukup	Tinggi	Cukup
175	Baik	Rendah	Cukup	Sangat Tinggi	Tinggi
176	Baik	Rendah	Baik	Sangat Rendah	Rendah
177	Baik	Rendah	Baik	Rendah	Cukup
178	Baik	Rendah	Baik	Cukup	Cukup
179	Baik	Rendah	Baik	Tinggi	Tinggi
180	Baik	Rendah	Baik	Sangat Tinggi	Tinggi
181	Baik	Cukup	Buruk	Sangat Rendah	Rendah
182	Baik	Cukup	Buruk	Rendah	Cukup
183	Baik	Cukup	Buruk	Cukup	Cukup
184	Baik	Cukup	Buruk	Tinggi	Tinggi
185	Baik	Cukup	Buruk	Sangat Tinggi	Tinggi

No	Variabel Input				Variabel Output
	Absensi	Prestasi	Disiplin	Loyalitas	Kinerja
186	Baik	Cukup	Cukup	Sangat Rendah	Cukup
187	Baik	Cukup	Cukup	Rendah	Tinggi
188	Baik	Cukup	Cukup	Cukup	Tinggi
189	Baik	Cukup	Cukup	Tinggi	Tinggi
190	Baik	Cukup	Cukup	Sangat Tinggi	Sangat Tinggi
191	Baik	Cukup	Baik	Sangat Rendah	Cukup
192	Baik	Cukup	Baik	Rendah	Tinggi
193	Baik	Cukup	Baik	Cukup	Tinggi
194	Baik	Cukup	Baik	Tinggi	Sangat Tinggi
195	Baik	Cukup	Baik	Sangat Tinggi	Sangat Tinggi
196	Baik	Tinggi	Buruk	Sangat Rendah	Cukup
197	Baik	Tinggi	Buruk	Rendah	Cukup
198	Baik	Tinggi	Buruk	Cukup	Cukup
199	Baik	Tinggi	Buruk	Tinggi	Tinggi
200	Baik	Tinggi	Buruk	Sangat Tinggi	Tinggi
201	Baik	Tinggi	Cukup	Sangat Rendah	Cukup
202	Baik	Tinggi	Cukup	Rendah	Cukup

No	Variabel Input				Variabel Output
	Absensi	Prestasi	Disiplin	Loyalitas	Kinerja
203	Baik	Tinggi	Cukup	Cukup	Tinggi
204	Baik	Tinggi	Cukup	Tinggi	Tinggi
205	Baik	Tinggi	Cukup	Sangat Tinggi	Tinggi
206	Baik	Tinggi	Baik	Sangat Rendah	Cukup
207	Baik	Tinggi	Baik	Rendah	Tinggi
208	Baik	Tinggi	Baik	Cukup	Sangat Tinggi
209	Baik	Tinggi	Baik	Tinggi	Sangat Tinggi
210	Baik	Tinggi	Baik	Sangat Tinggi	Sangat Tinggi
211	Baik	Sangat Tinggi	Buruk	Sangat Rendah	Cukup
212	Baik	Sangat Tinggi	Buruk	Rendah	Cukup
213	Baik	Sangat Tinggi	Buruk	Cukup	Tinggi
214	Baik	Sangat Tinggi	Buruk	Tinggi	Tinggi
215	Baik	Sangat Tinggi	Buruk	Sangat Tinggi	Tinggi
216	Baik	Sangat Tinggi	Cukup	Sangat Rendah	Cukup
217	Baik	Sangat Tinggi	Cukup	Rendah	Tinggi
218	Baik	Sangat Tinggi	Cukup	Cukup	Tinggi
219	Baik	Sangat Tinggi	Cukup	Tinggi	Sangat Tinggi

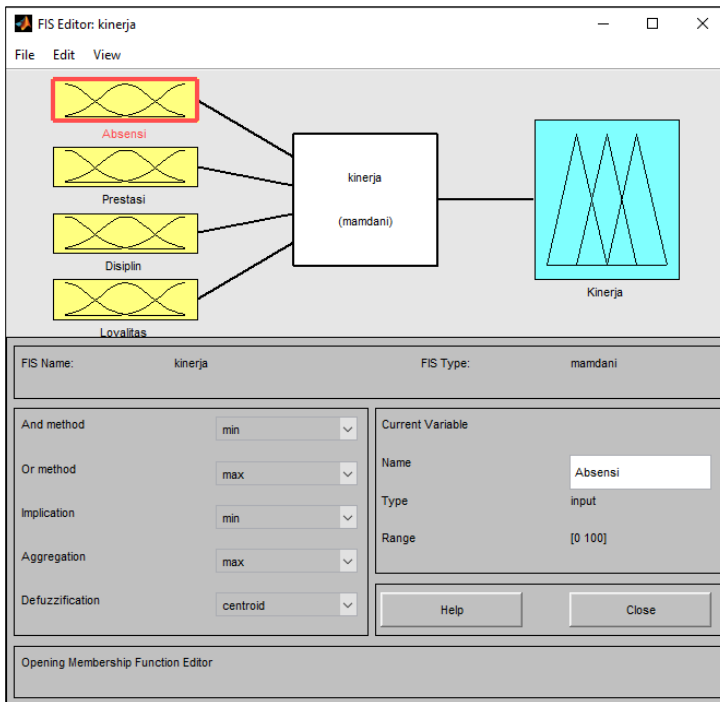


No	Variabel Input				Variabel Output
	Absensi	Prestasi	Disiplin	Loyalitas	Kinerja
220	Baik	Sangat Tinggi	Cukup	Sangat Tinggi	Sangat Tinggi
221	Baik	Sangat Tinggi	Baik	Sangat Rendah	Cukup
222	Baik	Sangat Tinggi	Baik	Rendah	Tinggi
223	Baik	Sangat Tinggi	Baik	Cukup	Sangat Tinggi
224	Baik	Sangat Tinggi	Baik	Tinggi	Sangat Tinggi
225	Baik	Sangat Tinggi	Baik	Sangat Tinggi	Sangat Tinggi

## H. Membangun Model Sistem Fuzzy

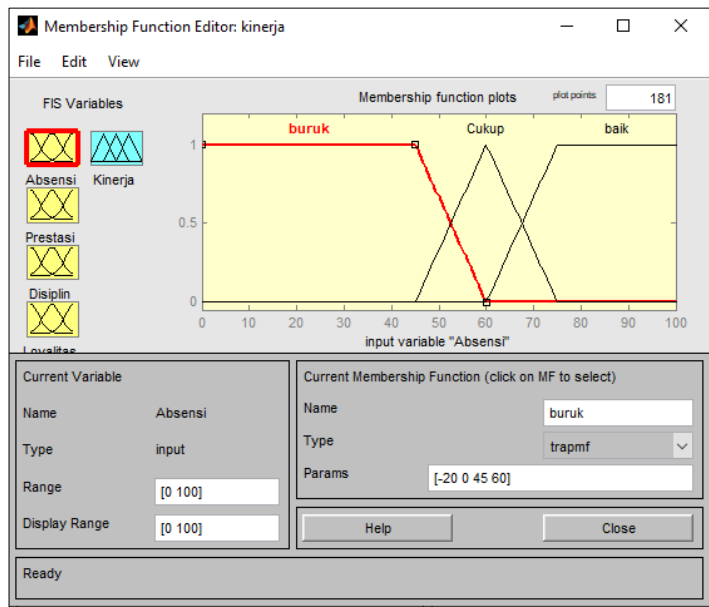
Setelah selesai dengan rancangan kurva fungsi keanggotaan dan juga rancangan rule base sistem fuzzy, maka tahap berikutnya adalah mengimplementasikan dan memasukan rancangan tersebut kedalam fuzzy toolbox yang ada di MATLAB, ada baiknya anda tidak melewatkan untuk coba mempraktekan latihan-latihan yang ada di chapter 2, karena pada bab ini saya hanya akan membahas poin-poin pentingnya saja secara garis besar, untuk detail langkah-langkah pengerjaannya anda bisa melihat pada chapter 2.

Silahkan anda masuk kedalam aplikasi matlab, kemudian buka fuzzy toolbox atau FIS editor, selanjutnya silahkan anda masukan rancangan kurva fungsi keanggotaan untuk setiap variabel input dan juga variabel output sesuai dengan desain dan rancangan yang telah dibuat, sehingga hasilnya akan tampak seperti gambar dibawah ini.

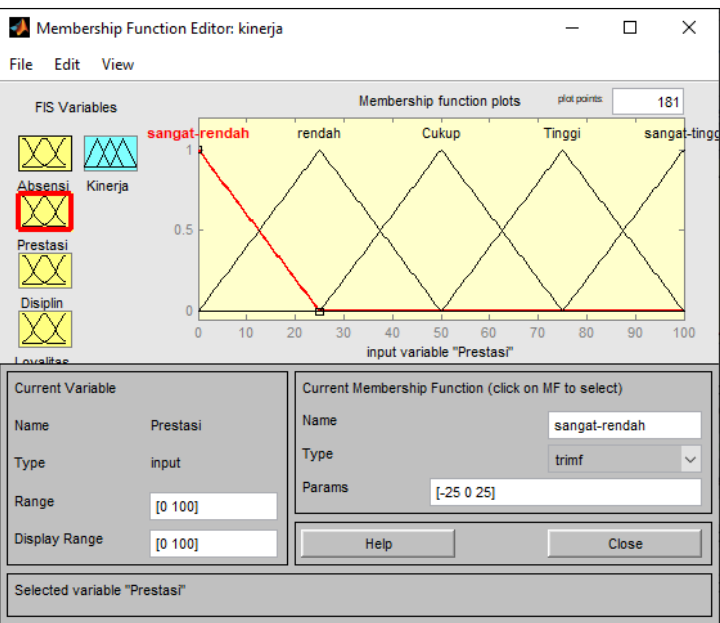


Dengan detail untuk setiap variabel adalah sebagai berikut:

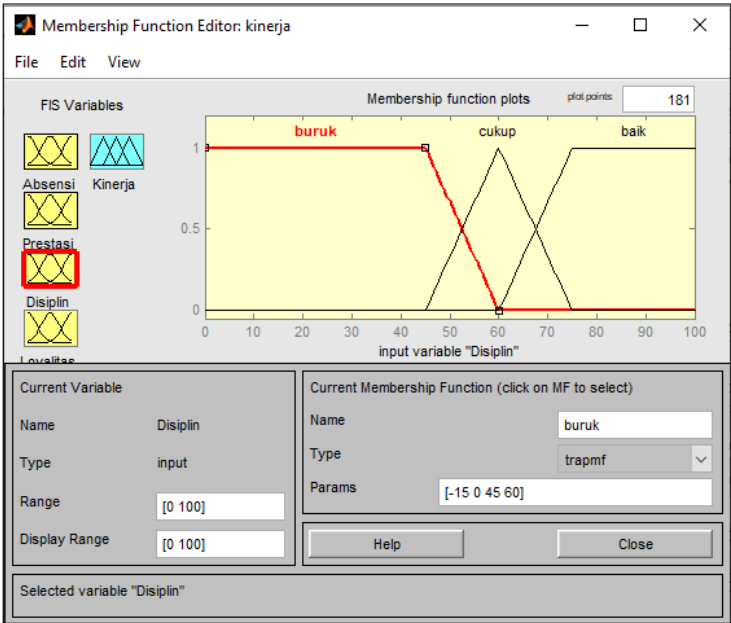
1. Kurva variabel input absensi



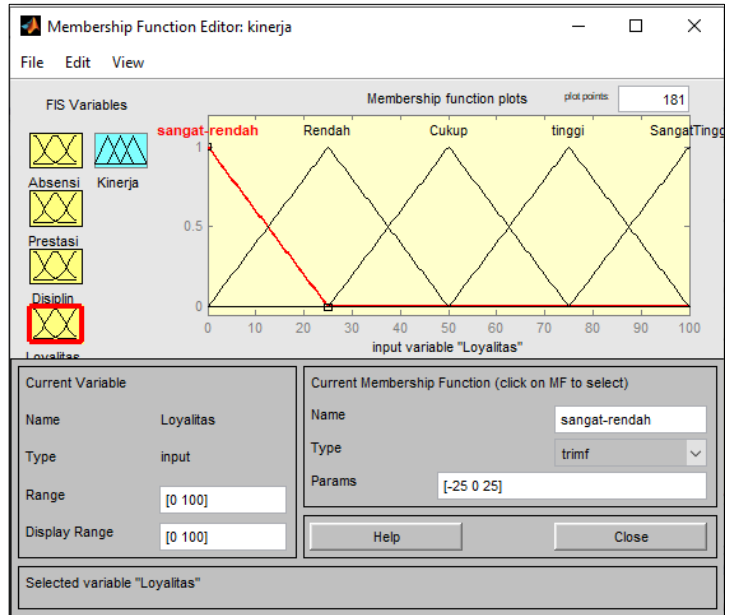
2. Kurva variabel input prestasi



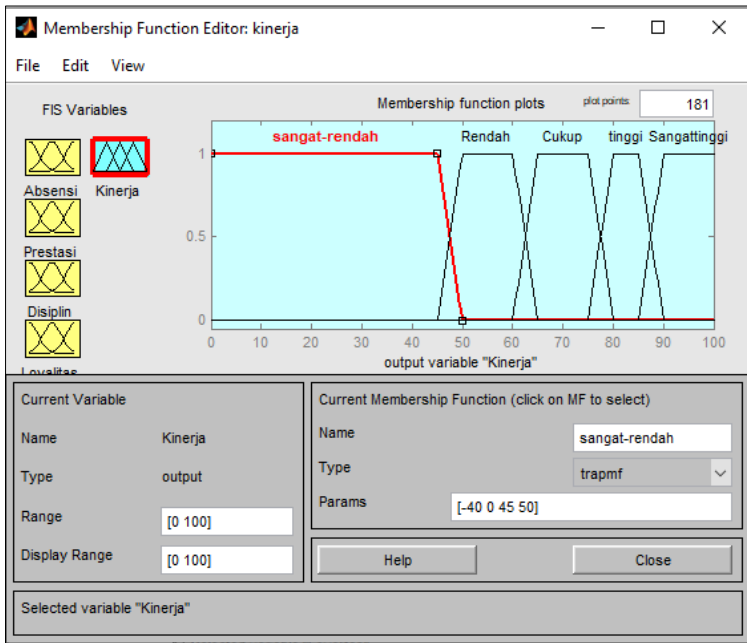
3. Kurva variabel input disiplin



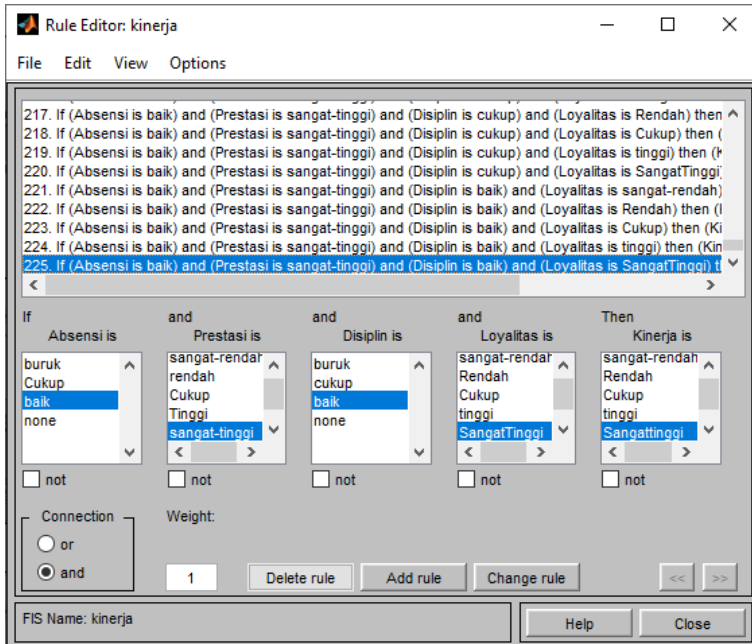
4. Kurva variabel input loyalitas



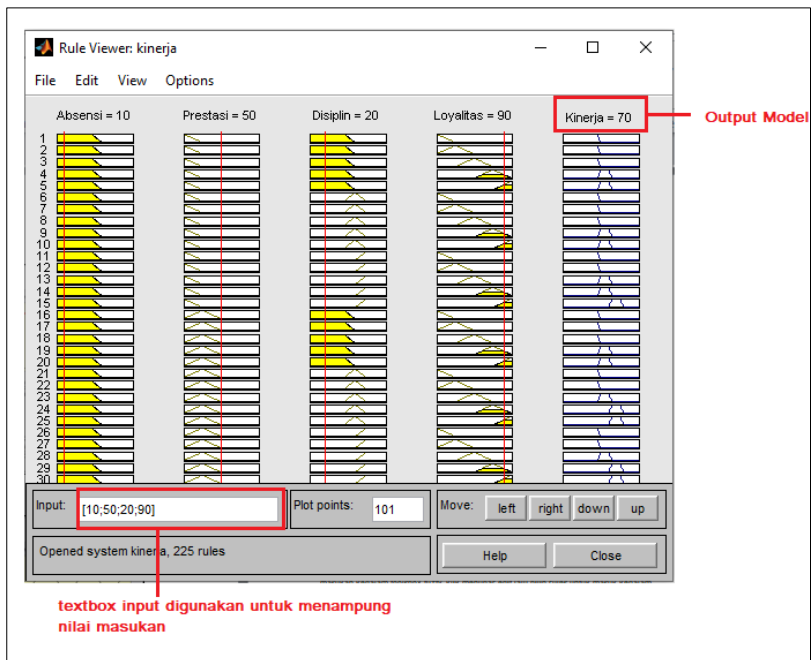
## 5. Kurva variabel output kinerja



- Setelah semua desain dan rancangan kurva fungsi keanggotaan selesai, maka tahap selanjutnya adalah memasukkan rule base sistem fuzzy yang telah kita buat sebelumnya, dimana pada rancangan tersebut terdapat sebanyak 255 rule base yang harus kita masukan kedalam toolbox fuzzy, klik menubar edit lalu pilih rules untuk masuk kedalam rule editor, kemudian masukan rulebase sesuai dengan rancangan yang telah dibuat, sehingga tampilan pada rule editor seperti pada gambar berikut:



- Setelah rulebase fuzzy selesai dimasukan, silahkan anda close, kemudian masuk kedalam menubar view lalu pilir rule untuk dapat masuk kedalam editor rule viewer, lakukan proses pengujian model fuzzy yang telah dibangun dengan memasukan nilai input pada komponen input di jendela editor rule viewer, perhatikan gambar dibawah ini



**Gambar 4.6** Tampilan Output Model Fuzzzy

Setelah itu simpanlah file model sisem fuzzy tersebut, dari menu file/export/to file, kemudian buatlah folder baru dengan nama kinerja karyawan, lalu didalam foler tersebut buat folder baru dengan nama **kinerja.fis**.

## I. Desain dan Rancangan Aplikasi GUI

### Desain dan Rancangan Basis Data

Basis data telah mempunyai bentuk normal selanjutnya dijelaskan dalam spesifikasi basis data yang meliputi file, media penyimpanan yang digunakan, isi yang disimpan, primary key, dan struktur file.

Buatlah database dengan nama kinerja, sesuaikan dengan rancangan dan struktur tabel didalam database tersebut

#### 1. Tabel Login

Tabel ini berfungsi untuk menampung data user sah yang mengelola data kebutuhan sistem. Primary Key (PK) dari

tabel ini adalah username. Untuk lebih jelasnya ditunjukkan pada tabel berikut ini:

**Tabel 4.2** Spesifikasi tabel Login

No.	Nama feild	Type	Size
1.	username (PK)	Varchar	20
2.	Password	Varchar	20

## 2. Tabel Karyawan

Tabel ini berfungsi untuk menampung data-data karyawan yang ditentukan dengan kriteria-kriteria yang ada. Untuk lebih jelasnya ditunjukkan pada tabel berikut:

**Tabel 4.3** Spesifikasi Tabel Karyawan

No.	Nama feild	Type	Size
1.	ID	Varchar	7
2.	NIK	Integer	15
3.	Nama	Varchar	25
4.	Alamat	Varchar	50
5.	Jenis Kelamin	Varchar	15
6.	Usia	Integer	3
7.	Pendidikan	Integer	3

## 3. Tabel Penilaian

Tabel ini berfungsi menampung data nilai setiap karyawan pada tiap. Untuk lebih jelasnya ditunjukkan pada tabel berikut ini:

**Tabel 4.4** Spesifikasi Tabel Penilaian

No.	Nama feild	Type	Size
1.	ID	Varchar	7
2.	id_Karyawan	Integer	3
3	Absensi	Integer	3
4	Prestasi	Integer	3
5	Disiplin	Integer	3
6	Loyalitas	Integer	3



No.	Nama feild	Type	Size
7	Skor Kinerja	float	-
8	Kategori	Varchar	20

### Desain dan Rancangan Antarmuka

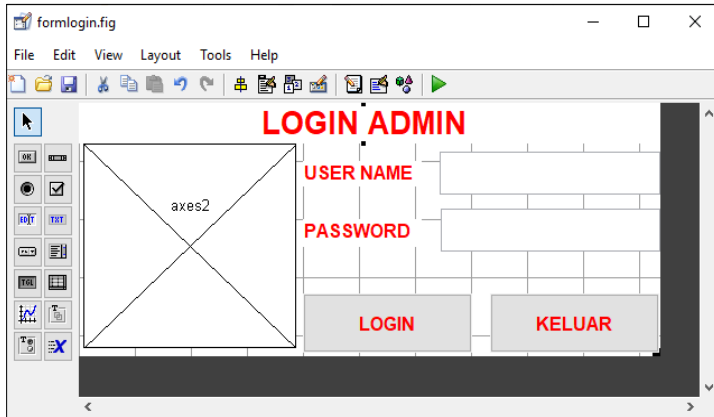
Antarmuka pengguna (User Interface) merupakan aspek sistem atau program yang dapat dilihat, komponen yang digunakan dan perintah-perintah yang digunakan untuk mengendalikan operasi dan masukan data. Desain dan rancangan antarmuka aplikasi penilaian kinerja karyawan digambarkan dalam bentuk yang jelas sebagai berikut:

**Tabel 4.5** Rancangan AntarMuka

Nama Form	Deskripsi
Form Login	Digunakan untuk melakukan autentifikasi user
Form Utama	Digunakan sebagai tampilan Menu yang terdiri dari submenu-submenu lainnya, dimana pengguna dapat menentukan kegiatan apa yang ingin dilakukan dalam sistem
Form Data Karyawan	Menu ini pengguna dapat memasukan, mengubah, menghapus, menyimpan serta mencetak data karyawan yang akan dinilai kinerjanya
Form Penilaian	Menu ini berfungsi untuk melakukan proses penlaian dan perhitungan sesuai model sistem fuzzy.

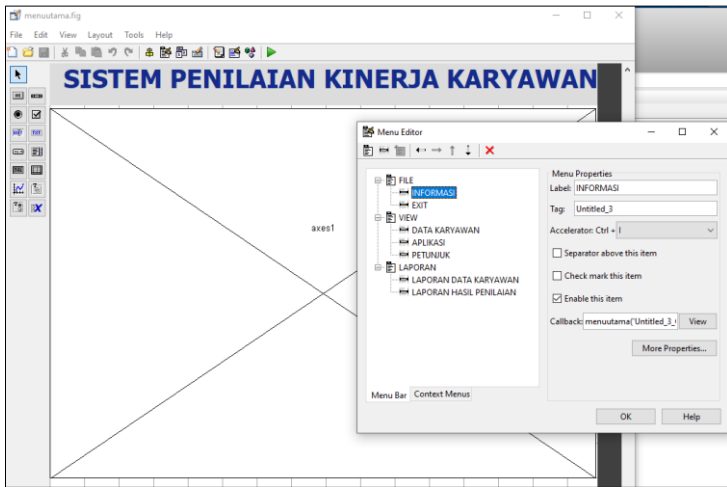
Dari tabel diatas terlihat bahwa aplikasi ini terdiri dari 4 form atau halaman, dengan detail rancangannya adalah:

1. Form login



Buatlah form login ini dengan menggunakan 3 buah statictext yang akan digunakan untuk label, 1 buah komponen axes yang akan digunakan untuk menampilkan logo/gambar pada form login, 2 buah edittext untuk menampung input username dan password dari pengguna dan 2 button yang akan digunakan untuk proses login dan proses keluar dari form login, kemudian simpan file didalam folder kinerja karyawan dengan nama formlogin.

## 2. Form Utama



Buatlah form menu utama ini dengan menggunakan 1 buah komponen axes yang akan digunakan untuk menampilkan logo/gambar pada form utama, 1 buah statictext yang akan digunakan untuk label atau judul aplikasi dan 3 buah menubar untuk menampilkan halaman yang dipilih pengguna pada submenu tersebut, kemudian simpan file didalam folder kinerja karyawan dengan nama menuutama.

### 3. Form Data Karyawan

The screenshot shows a Java Swing window titled "datakaryawan.fig". The window has a menu bar with "File", "Edit", "View", "Layout", "Tools", and "Help". Below the menu bar is a toolbar with various icons. The main content area is titled "DATA KARYAWAN" in blue text. On the left side, there is a vertical toolbar with icons for file operations, editing, and viewing. The form itself consists of several input fields and buttons. The input fields are labeled "NIK", "NAMA", "JENIS KELAMIN", "ALAMAT", "USIA", and "PENDIDIKAN". The "JENIS KELAMIN" and "PENDIDIKAN" fields have dropdown menus with the text "PILIH...!". To the right of the input fields, there are five buttons: "TAMBAH", "HAPUS", "SIMPAN", "LAPORAN", and "KELUAR". At the bottom of the form, there is a table with the following columns: "ID", "NIK", "NAMA", "JENIS KELAMIN", "ALAMAT", "USIA", and "PENDIDIKAN". The table has four rows, with the first row containing the numbers 1 through 4 in the "ID" column.

ID	NIK	NAMA	JENIS KELAMIN	ALAMAT	USIA	PENDIDIKAN
1						
2						
3						
4						

Buatlah form data karyawan dengan menggunakan 3 buah panel yang akan digunakan untuk mengelompokkan komponen lain menjadi 3 kelompok, dimana panel1 berisi komponen statictext yang akan digunakan untuk label atau judul aplikasi 6 buah, kemudian panel2 berisi 5 buah komponen button yang berfungsi untuk menambah, menghapus, menyimpan, dan keluar dari form data karyawan, dan panel 3 berisi 1 buah komponen uitable yang akan digunakan untuk menampilkan data karyawan dari database, kemudian simpan file didalam folder kinerja karyawan dengan nama datakaryawan.

#### 4. Form Penilaian (aplikasi)

The screenshot shows a MATLAB application window titled "aplikasi.fig". The window contains a form titled "APLIKASI PENILAIAN KARYAWAN". The form has a left sidebar with icons for various functions. The main area is divided into several sections:

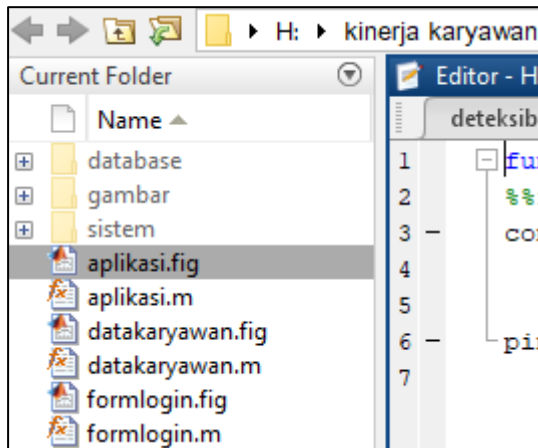
- Input Fields:** Fields for NIK, NAMA, ABSENSI, PRESTASI, DISIPLIN, and LOYALITAS, each with a corresponding input box. A "CARI" button is next to the NIK field.
- SKOR NILAI:** A section with a large empty box for the score and a "STATUS KINERJA" label above a red box.
- Action Buttons:** Four buttons labeled "PROSES", "SIMPAN", "PETUNJUK", and "KELUAR" are arranged horizontally.
- Table:** A table at the bottom with columns: NIK, NAMA, ABSENSI, PRESTASI, DISIPLIN, LOYALITAS, SKOR, and KINERJA. It has rows numbered 1 to 4.

Buatlah form aplikasi dengan menggunakan komponen seperti gambar diatas, komponen-komponen yang digunakan hamper serupa dengan form data karyawan, kemudian simpan file didalam folder kinerja karyawan dengan nama aplikasi.

#### J. Membangun Aplikasi Penilaian Kinerja

Pada bagian ini kita akan mulai membangun aplikasi dengan menggunakan bahasa pemrograman matlab, sesuai dengan rancangan antar muka yang telah direncanakan sebelumnya, maka kita akan mulai membuat fungsi dan memasukan proses Logika dan sourcecode sampai dengan pengujian setiap form yang akan kita buat. **Pastikan anda telah membuat dummy data didalam database kinerja, dan juga**

anda telah membuat folder gambar yang berisi gambar-gambar yang akan digunakan di aplikasi yang akan dibangun.



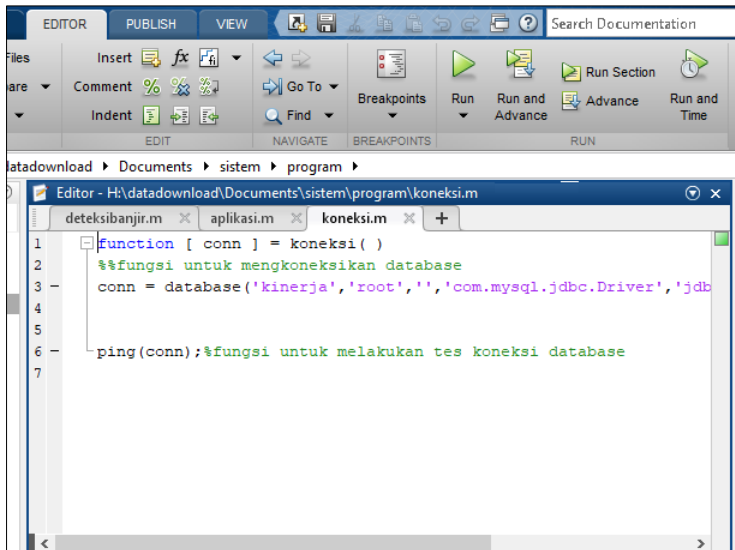
**Gambar 4.7** Struktur Folder Aplikasi Penilaian Kinerja

### Membuat Fungsi khusus

Pada bagian ini kita akan membuat file dengan fungsi khusus atau tersendiri tanpa menggunakan form atau desain GUI, ini bertujuan agar kita tidak perlu membuat source code dengan fungsi yang sama secara berulang, kita hanya perlu memanggil fungsi-fungsi khusus ini saja. Adapun fungsi khusus yang akan kita buat adalah:

#### 1. Fungsi koneksi database

Pastikan kita telah membuat database sesuai dengan rancangan database yang telah dibahas sebelumnya. Buatlah file baru pada tab editor m.file, perhatikan gambar dibawah ini



dan simpan didalam folder kinerja karyawan dengan nama koneksi.m, kemudian masukan sourcecode berikut

```
function [ conn ] = koneksi()
conn = database('kinerja','root','...',
'com.mysql.jdbc.Driver','jdbc:mysql://localhost:3306/');
ping(conn);
```

penjelasan code:  
function [ conn ]; deklarasi fungsi koneksi  
conn = database; fungsi untuk mengkoneksikan database,  
perhatikan nama DB yang kita panggil dan port dari  
Mysql kita , defaultnya adalah 3306, jika kita telah  
mengubahnya maka silahkan sesuaikan  
ping(conn); fungsi untuk melakukan tes koneksi database

Silahkan running fungsi koneksi database tersebut, pastikan kita telah mengaktifkan database pada control panel DBMS kita (Xampp), maka output dari fungsi ini bisa kita lihat pada command windows seperti gambar dibawah ini;

```

Command Window

>> koneksi

ans =

    Instance: 'karyawan'
    UserName: 'root'
    Driver: 'com.mysql.jdbc.Driver'
    URL: 'jdbc:mysql://localhost:3306/'
    Constructor: [1x1 com.mathworks.toolbox.database.databaseConnect]
    Message: []
    Handle: [1x1 com.mysql.jdbc.JDBC4Connection]
    Timeout: 0
    AutoCommit: 'on'
    Type: 'Database Object'

fx >>

```

**Gambar 4.8** Tampilan Fungsi Koneksi Database

## 2. Fungsi Laporan Data karyawan

Fungsi ini dibuat untuk mencetak dat karyawan dalam bentuk dokumen jika memang diperlukan. Fungsi ini akan digunakan di form yang berbeda, yaitu form utama dan juga form data karyawan. Buatlah file baru dengan nama laporan.m, kemudian masukan source kode dibawah ini

```

function laporan()
f = figure('Position', [100 100 860 600])%memanggil form baru
annotation(f,'textbox',...
[0.0285 0.91551724137931 0.93025 0.0586206896551724],...
'String',{'LAPORAN DATA KARYAWAN'},...
'HorizontalAlignment','center',...
'FontWeight','bold',...
'FontSize',18,...
'FitBoxToText','off');
t = uitable('Parent', f, 'Position', [25 25 810 500]);%membuat
tabel baru
set(t, 'ColumnName', {'Id', 'NIK', 'Nama', 'Jenis | kelamin',
'Alamat', 'Usia', 'Pendidikan'});
set(t, 'ColumnWidth', {50 100 100 100 150 100 100});

disp('Connecting to MySQL');%melakukan koneksi database
conn = koneksi();

```



```

tablename = 'karyawan';%memanggil data dari tabel
perusahaan
colnames = {'Id', 'NIK', 'Nama', 'Jenis_kelamin', 'Alamat',
'Usia', 'Pendidikan'};
h = exec(conn,'select * from karyawan');%mengeksekusi
perintah SQL
h = fetch(h);
data = h.Data;
set(t,'Data',data);%menempatkan data dari tabel perusahaan
kedalm tabel baru
close(conn);%menutup koneksi database

```

Silahkan jalankan file tersebut, maka tampilan akan seperti gambar dibawah ini

	Id	NIK	Nama	Jenis kelamin	Alamat	Usia	Pendidikan
1	1	20100101	Jamal	Pria	Tangerang	28	S1
2	2	20100102	Riani	Wanita	Jakarta	25	SMA
3	4	20100104	Muhamad alamsyah	pria	Jakarta	32	SMA
4	5	20100103	Maryam	Wanita	Jakarta	25	D3
5	6	20100106	Kasiman	pria	Tangerang	32	S1
6	7	20100107	Kamil	pria	Jakarta	32	D3
7	8	20100108	Jaka	pria	Tangerang	32	D3
8	9	20100109	Lidia	Wanita	Tangerang	32	SMA

**Gambar 4.9** Tampilan Laporan Data Karyawan

Fungsi tersebut akan mengambil data karyawan yang tersimpan didalam database dan dapat dicetak kedalam bentuk dokumen .

### 3. Fungsi Laporan hasil Penilaian

Fungsi ini hampir sama dengan fungsi laporan data karyawan, hanya saja ini menampilkan hasil penilaian kinerja dari seluruh karyawan yang telah diproses dengan model sistem fuzzy. Buatlah file baru dan simpan dengan nama laporanhasil.m, kemudian masukan sourcecode berikut ini:

```
function laporanhasil()
f = figure('Position', [100 100 860 600])%memanggil form baru
annotation(f,'textbox',...
[0.0285 0.91551724137931 0.93025 0.0586206896551724],...
'String',{'LAPORAN PENILAIAN KINERJA'},...
'HorizontalAlignment','center',...
'FontWeight','bold',...
'FontSize',18,...
'FitBoxToText','off');
t = uitable('Parent', f, 'Position', [25 25 810 500]);%membuat
tabel baru
set(t, 'ColumnName', {'Id', 'NIK', 'Nama', 'Absensi', 'Prestasi',
'Disiplin', 'Loyalitas', 'Skor|Nilai', 'Kinerja'});
set(t, 'ColumnWidth', {50 70 100 80 80 80 80 80 100});
disp('Connecting to MySQL');%melakukan koneksi database
conn = koneksi();
tablename = 'nilai_kinerja';%memanggil data dari tabel
perusahaan
colnames = {'Id', 'NIK', 'Nama', 'Absensi', 'Prestasi', 'Disiplin',
'Loyalitas', 'Skor_Nilai', 'Kinerja'};
h = exec(conn,'select * from nilai_kinerja');%mengeksekusi
perintah SQL
h = fetch(h);
data = h.Data;
set(t,'Data',data);%menempatkan data dari tabel perusahaan
kedalm tabel baru
close(conn);%menutup koneksi database
```

Selanjutnya silahkan coba jalankan file tersebut, maka hasilnya akan seperti gambar dibawah ini



	Id	NIK	Nama	Absensi	Prestasi	Disiplin	Loyalitas	Skor Nilai	Kinerja
1		120100101	Jamal	80	60	60	40	76.1364	CUKUP
2		2							
3		3							
4		4							

**Gambar 4.10** Tampilan Laporan Hasil Penilaian

**Memasukan Source Code Setiap Form**

Pada tahap ini kita akan memasukan sourcecode kedalam setiap fom pada aplikasi ini, yang penting untuk anda fahami dan ingat adalah tag name setiap komponen didalam form yang akan kita gunakan.

**1. Source Code Form Login**

Form login merupakan form yang akan muncul pertama kali ketika pengguna masuk kedalam aplikasi peilaian kinerja karyawan, form ini berfungsi untuk melakukan autentikasi pengguna yang akan masuk kedalam sistem.

Pada desain dan rancangan form login diatas, terdapat 4 buah .komponen penting yang akan digunakan sesuai dengan alur procedural dan logika, yaitu 2 komponen

edittext yang akan digunakan untuk menampung input username dan password dari pengguna dan 2 komponen button yang akan berfungsi untuk melakukan proses autentikasi atau validasi dari username dan password yang telah di masukan pengguna, serta button yang berfungsi untuk keluar dari aplikasi. Oleh karena itu sangat penting untuk memastikan penggunaan tag name pada 4 komponen tersebut karena kita akan memasukan sourcecode kedalamnya.

Silahkan anda masukan sourcecode dibawah ini dan perhatikan tag name pada fungsi komponen tersebut, pastikan anda sudah membuat dummy data untuk user, dan juga menyiapkan gambar yang akan ditampilkan untuk form login dan form lainnya yang tersimpan didalam folder kinerja karyawan/gambar.

<p><b>1. Sourcecode pada bagian opening function (line 48)</b></p> <pre>function formlogin_OpeningFcn(hObject, eventdata, handles, varargin) handles.output = hObject;  % fungsi untuk menampilkan gambar guidata(hObject, handles); x=imread('gambar/D.jpg'); axes(handles.axes2) imshow(x);</pre>
<p><b>2. Sourcecode Edittext2 (password)</b></p> <p>- Untuk komponen ini menggunakan fungsi KeyPress untuk memasukan sourcecodenya, caranya dengan klikkanan didalam komponen tersebut, lalu pilih view callbacks/keypressfcn</p> <pre>function edit2_KeyPressFcn(hObject, eventdata, handles) % mengubah input password menjadi tersandi password = get(handles.edit2,'userdata'); key = get(handles.output,'currentkey'); switch key case 'backspace'</pre>

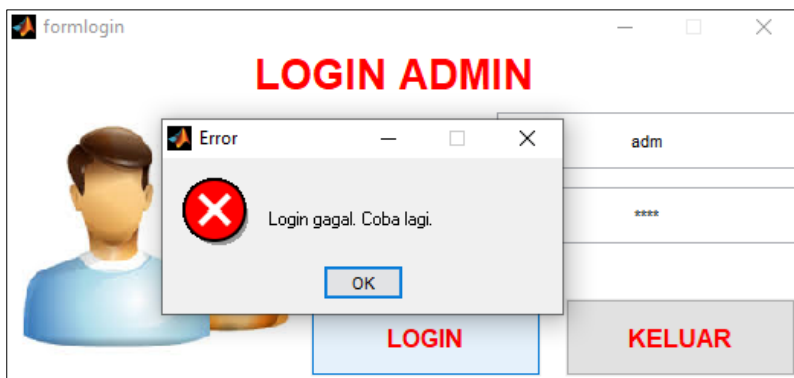
```

password = password(1:end-1);
SizePass = size(password);
if SizePass(2) > 0
    asterisk(1,1:SizePass(2)) = '•';
    set(handles.edit2,'String',asterisk)
else
    set(handles.edit2,'String','')
end
set(handles.edit2,'Userdata',password)
case 'escape'
case 'insert'
case 'delete'
case 'home'
case 'pageup'
case 'pagedown'
case 'end'
case 'rightarrow'
case 'downarrow'
case 'leftarrow'
case 'uparrow'
case 'shift'
case 'return'
case 'alt'
case 'control'
case 'windows'
otherwise
    password = [password
get(handles.output,'currentcharacter')];
    SizePass = size(password);
    if SizePass(2) > 0
        asterisk(1:SizePass(2)) = '*';
        set(handles.edit2,'string',asterisk)
    else
        set(handles.edit2,'String','');
    end
    set(handles.edit2,'Userdata',password) ;
end
end

```

3. Sourcecode button1 (Login)
<pre> <b>function</b> pushbutton1_Callback(hObject, eventdata, handles) % mengambil nilai username dan password UN = get(handles.edit1,'string'); ID = get(handles.edit2,'userdata');  % Lakukan pengecekan login ke database query = sprintf('SELECT * FROM user WHERE username = "%s" AND password = "%s"', UN, ID); results = fetch(koneksi, query);  % Periksa apakah login berhasil <b>if</b> ~isempty(results) % Jika berhasil, tampilkan pesan berhasil close(formlogin); guidata(menuutama); <b>else</b> % Jika gagal, tampilkan pesan gagal msgbox('Login gagal. Coba lagi.', 'Error', 'error', 'modal'); <b>end</b> </pre>
4. Sourcecode button 2(keluar)
<pre> <b>function</b> pushbutton2_Callback(hObject, eventdata, handles) % menutup form atau keluar dari form close; </pre>

Setelah selesai silahkan anda running form login, dan lakukan pengujian terhadap form login, perhatikan apakah sudah berjalan dengan semestinya dan sesuai dengan fungsinya



**Gambar 4.11** Tampilan Form Login

Pada form login jika username dan password sesuai dengan yang di database, maka pengguna akan masuk kedalam menu utama, tetapi jika tidak sesuai maka sistem akan menampilkan pesan error, seperti pada gambar di atas.

## 2. SourceCode Form Menuutama

Pada Desain dan rancangan menu utama terdapat 2 komponen saja, yaitu axes yang berguna untuk menampilkan gambar background menu utama dan komponen menubar dan sub menubar yang berfungsi untuk mengarahkan kehalaman lain sesuai dengan input dari pengguna, silahkan masukan sourcecode berikut ini.

### 1. Pada bagian opening function line 48.

```
function menuutama_OpeningFcn(hObject, eventdata, handles, varargin)
```

```
handles.output = hObject;
```

```
guidata(hObject, handles);
```

```
x=imread('image/logo.jpg'); % memanggil gambar yang berada di folder image
```

```
axes(handles.axes1); %fungsi untuk menempatkan gambar opada komponen axes
```

```
imshow(x); %fungsi menampilkan gambar
```

### 2. Pada komponen submenu Laporan karyawan

<b>function Untitled_14_Callback(hObject, eventdata, handles)</b> % memanggil fungsi laporan.m laporan;
<b>3. Pada komponen submenu Laporan Penilaian</b>
<b>function Untitled_15_Callback(hObject, eventdata, handles)</b> % memanggil fungsi laporan hasil.m laporanhasil;
<b>4. Pada komponen submenu Exit</b>
<b>function Untitled_8_Callback(hObject, eventdata, handles)</b> % Menampilkan dialogbox untuk keluar dari aplikasi selection=questdlg(['Anda Yakin Ingin Keluar','?'],... ['Keluar' ' '...'],... 'Ya','tidak','Ya'); if strcmp(selection,'tidak') return end close;
<b>5. Pada komponen submenu Data Karyawan</b>
<b>function Untitled_10_Callback(hObject, eventdata, handles)</b> % memanggil form data karyawan datakaryawan;
<b>6. Pada komponen submenu Aplikasi</b>
<b>function Untitled_5_Callback(hObject, eventdata, handles)</b> % memanggil form aplikasi aplikasi;

Lakukanlah proses testing dan pengujian pada form yang telah dibuat dan dimasukan source code pada komponen-komponen ataupun fungsi lainnya. Silahkan Jalankan form menu utama, dan uji fungsi opening function yang berguna untuk menampilkan gambar, uji fungsi setiap komponen dari sub menubar yang tersedia.





**Gambar 4.12** Tampilan Form Utama

Pada gambar di atas adalah tampilan Menu utama pada sistem penilaian kinerja karyawan, dimana pada halaman ini pengguna aplikasi menentukan aktifitas sesuai dengan submenu yang ada didalamnya.

### 3. Sourcecode Form Data Karyawan

Berdasarkan pada desain dan rancangan pada halaman data karyawan, terdapat banyak komponen yang digunakan sebagai berikut:

#### 1. Opening Function

```
function datakaryawan_OpeningFcn(hObject, eventdata,
handles, varargin)
% menampilkan data karyawan saat for data karyawan
muncul
handles.output = hObject;
guidata(hObject, handles);
disp('Connecting to MySQL'); % mengkoneksikan ke
database mySQL
conn = koneksi();
```

<pre> h = exec(conn,'select * from karyawan'); h = fetch(h); data = h.Data; set(handles.uitable1,'Data',data) </pre>
<p>2. Menu PopUp Jenis kelamin</p> <pre> function jk_Callback(hObject, eventdata, handles) % Memasukanpilihan jenis kelamin di menu popup jk=get(hObject,'value'); switch jk case 1 valuejk=""; case 2 valuejk='pria'; case 3 valuejk='Wanita'; end handles.valuejk= valuejk; guidata(hObject, handles); </pre>
<p>3. Menu PopUp Pendidikan</p> <pre> function pend_i_Callback(hObject, eventdata, handles) % Menampilkan Pilihan Menu popup pendidikan pendi=get(hObject,'value'); switch pend_i case 1 valuepdd=""; case 2 valuepdd='SMA'; case 3 valuepdd='D3'; case 4 valuepdd='S1'; end handles.valuependi= valuepdd; guidata(hObject, handles); </pre>
<p>4. Button Hapus data</p> <pre> function hapus_Callback(hObject, eventdata, handles) % mengambil nilai berdasarkan NIK </pre>

```

nik = get(handles.nik,'String');
%menghapus data pada database berdasarkan NIK
sql1=['delete from karyawan where NIK = ' '' nik '''];
sql2=['delete from nilai_kinerja where NIK = ' '' nik '''];
disp('Connecting to MySQL');
conn = koneksi();
curs = exec(conn,sql1);
curs = exec(conn,'select * from karyawan');
curs = fetch(curs);
data = curs.Data;
set(handles.uitable1,'Data',data)

conn = koneksi();
curs = exec(conn,sql2);
curs = exec(conn,'select * from nilai_kinerja');
curs = fetch(curs);
data = curs.Data;
disp(data)

```

#### 5. Button Tambah data

```

function input_Callback(hObject, eventdata, handles)
% mengaktifkan komponen-komponen lain di form
proyek=guidata(gcbo);
set(proyek.jk,'enable','on');
set(proyek.nik,'enable','on');
set(proyek.nama,'enable','on');
set(proyek.usia,'enable','on');
set(proyek.almt,'enable','on');
set(proyek.pendi,'enable','on');
set(proyek.simpan,'enable','on');

```

#### 6. Button Simpan data

```

function simpan_Callback(hObject, eventdata, handles)
% mengambil nilai dari setiap komponen input
proyek=guidata(gcbo);
bersih="";
nik= get(handles.nik,'String');
nama=get(handles.nama,'String');

```

```

alamat=get(handles.almt,'String');
usia=get(handles.usia,'String');
JK=handles.valuejk;
pdd=handles.valuependi;
%elakukan validasi input NIK
nik= str2double(nik);
if isnan(nik)
    msgbox('input hanya berupa
angka...!!!','WARNING...!!!','warn','modal')
    set(handles.nik,'string','')
    return
end
% mengkoneksikan ke database mySQL
conn = koneksi();
curs = exec(conn,'select * from karyawan');%memanggil
semua data dari database
curs = fetch(curs);
curs.Data

tablename = 'karyawan';
colnames = {'NIK', 'Nama', 'Jenis_kelamin', 'Alamat',
'Usia', 'Pendidikan'};
data1 = {nik nama JK alamat usia pdd}
insert(conn,'karyawan',{'NIK', 'Nama', 'Jenis_kelamin',
'Alamat', 'Usia', 'Pendidikan'},...
{nik nama JK alamat usia pdd});
h = exec(conn,'select * from karyawan');
h = fetch(h);
data1 = h.Data;
% memasukan data ke dalam tabel
set(handles.uitable1,'Data',data1);
close(conn);
% membersihkan semua komponen di didalam form
set(proyek.nama,'string',bersih);
set(proyek.nik,'string',bersih);
set(proyek.almt,'string',bersih);
set(proyek.usia,'string',bersih);

```

```
set(handles.jk,'value',1);
set(handles.pendi,'value',1);

7. Button Laporan
function cetak_Callback(hObject, eventdata, handles)
% memanggil fungsi untuk membuat laporan data
karyawan
laporan;

8. Button Keluar
function keluar_Callback(hObject, eventdata, handles)
% menutup form data karyawan dan kembali ke form
menu utama
close;
menuutama;
```

Jalankan form data karyawan, dan perhatikan komponen apa saja yang kondisinya dalam keadaan aktif dan tidak aktif, seperti gambar di bawah ini

datakaryawan

DATA KARYAWAN

NIK

NAMA

JENIS KELAMIN

PILIH...!

ALAMAT

USIA

PENDIDIKAN

PILIH...!

TAMBAH

HAPUS

SIMPAN

LAPORAN

KELUAR

	ID	NIK	NAMA	JENIS KELAMIN	ALAMAT	USIA	PENDIDIKAN
1	1	20100101	Jamal	Pria	Tangerang	28	S1
2	2	20100102	Riani	Wanita	Jakarta	25	SMA
3	4	20100104	Muhamad ala...	pria	Jakarta	32	SMA
4	5	20100103	Maryam	Wanita	Jakarta	25	D3
5	6	20100106	Kasiman	pria	Tangerang	32	S1
6	7	20100107	Kamil	pria	Jakarta	32	D3
7	8	20100108	Jaka	pria	Tangerang	32	D3
8	9	20100109	Lidia	Wanita	Tangerang	32	SMA

Saat form ini dijalankan, maka hanya dua tombol yang akan aktif, sehingga aktifitas dari user pertama kali adalah tombol hapus, tombol tambah, tombol laporan dan tombol keluar. user dapat menghapus data berdasarkan dari NIK karyawan, selanjutnya jika ingin menambahkan data karyawan user dapat menggunakan button tambah untuk mengaktifkan komponen lainnya.

datakaryawan

DATA KARYAWAN

NIK

2010176

NAMA

Kartika

JENIS KELAMIN

Wanita

ALAMAT

Jakarta Utara

USIA

35

PENDIDIKAN

D3

TAMBAH

HAPUS

SIMPAN

LAPORAN

KELUAR

	ID	NIK	NAMA	JENIS KELAMIN	ALAMAT	USIA	PENDIDIKAN
1	1	20100101	Jamal	Pria	Tangerang	28	S1
2	2	20100102	Riani	Wanita	Jakarta	25	SMA
3	4	20100104	Muhamad ala...	pria	Jakarta	32	SMA
4	5	20100103	Maryam	Wanita	Jakarta	25	D3
5	6	20100106	Kasiman	pria	Tangerang	32	S1
6	7	20100107	Kamil	pria	Jakarta	32	D3
7	8	20100108	Jaka	pria	Tangerang	32	D3
8	9	20100109	Lidia	Wanita	Tangerang	32	SMA

Gambar 4.13 Tampilan Form Data Karyawan

Pada form ini ketika kita klik tombol keluar maka akan muncul kotak dialog konfirmasi, jika pilihan adalah tidak maka sistem akan tetap berada di form data karyawan, sedangkan jika pilihan adalah ya, maka sistem akan menutup form data karyawan dan akan kembali ke menu utama. Lakukanlah pengujian secara menyeluruh terhadap semua fungsi pada form data karyawan.

#### 4. Sourcecode Aplikasi Penilaian Kinerja

Pada studi kasus ini, inti atau bagian utama dari aplikasi penilaian kinerja karyawan ini adalah Form Penilaian kinerja karyawan, dimana pada form ini kita akan mengintegrasikan model sistem fuzzy yang telah kita buat kedalam aplikasi penilaian kinerja, sehingga proses penilaian akan dilakukan dengan sistem yang berdasarkan pengetahuan (artificial intelligence)

Sesuai dengan desain dan rancangan form aplikasi penilaian, silahkan masukan sourcecode berikut ini pada komponen button di form tersebut.

##### 1. Sourcecode Button Cari

```
function pushbutton6_Callback(hObject, eventdata, handles)
% mengambil nilai NIK
nik = get(handles.nik,'String');
% mencari data nik pada database
conn=koneksi();
sql=['select * from karyawan where NIK= ' '' nik '''];
h = exec(conn,'select * from karyawan');

data=fetch(conn,sql);
if isempty(data)
    msgbox('data tdk ditemukan')

    set(handles.nama,'string','')
else
    set(handles.nama,'string',data(1,3))
    proyek=guidata(gcbo);
    set(proyek.absensi,'enable','on');
    set(proyek.dis,'enable','on');
    set(proyek.prestasi,'enable','on');
    set(proyek.loyalitas,'enable','on');
end
```

##### 2. Sourcecode button proses

```
function pushbutton1_Callback(hObject, eventdata, handles)
% mengambil nilai input pada komponen edittext
```

```

a = get(handles.absensi,'string');
b = get(handles.prestasi,'string');
c = get(handles.dis,'string');
d = get(handles.loyalitas,'string');

A=readfis('sistem/kinerja');
out=evalfis([a b c d],A);
set(handles.nilai,'string',out)
if(out<43)
    msgbox('Status karyawan masuk kedalam grade (kategori)
SANGAT RENDAH','pemberitahuan')
    set(handles.grade,'string','SANGAT RENDAH')
elseif(out<63)
    msgbox('Status karyawan masuk kedalam grade (kategori)
RENDAH','pemberitahuan')
    set(handles.grade,'string','RENDAH')
elseif(out<78)
    msgbox('Status karyawan masuk kedalam grade (kategori)
CUKUP','pemberitahuan')
    set(handles.grade,'string','CUKUP')
elseif(out<87)
    msgbox('Status karyawan masuk kedalam grade (kategori)
TINGGI','pemberitahuan')
    set(handles.grade,'string','TINGGI')
else
    msgbox('Status karyawan masuk kedalam grade
(kategori)SANGAT TINGGI','pemberitahuan')
    set(handles.grade,'string','SANGAT TINGGI')
end
global r
nik= get(handles.nik,'string');
nama = get(handles.nama,'string');
x=num2str(nik);
x1=cell2mat(nama);
x2=num2str(x1)
set(handles.nik,'string',x);
set(handles.nama,'string',x2);

```



```

A= get(handles.nik,'string');
B = get(handles.nama,'string');
C = get(handles.absensi,'string');
D = get(handles.prestasi,'string');
E = get(handles.dis,'string');
F = get(handles.loyalitas,'string');
G = get(handles.nilai,'string');
H = get(handles.grade,'string');
r = r+1;
z = get(handles.uitable1,'Data');
z(r,1:8) = { A B C D E F G H};
set(handles.uitable1,'Data',z)
proyek=guidata(gcbo);
pesan=";
set(proyek.nik,'string',pesan);
set(proyek.nama,'string',pesan);
set(proyek.absensi,'string',pesan);
set(proyek.prestasi,'string',pesan);
set(proyek.dis,'string',pesan);
set(proyek.loyalitas,'string',pesan);
set(proyek.nilai,'string',pesan);
set(proyek.grade,'string',pesan);

```

### 3. Sourcecode button simpan

```

function pushbutton3_Callback(hObject, eventdata, handles)
% memanggil semua data dari database seleksi
conn = koneksi();
curs = exec(conn,'select * from nilai_kinerja');
exec(conn,'TRUNCATE TABLE nilai_kinerja');
data=get(handles.uitable1,'Data');
tablename = 'nilai_kinerja';
colnames = {'NIK', 'Nama', 'Absensi', 'Prestasi', 'Disiplin',
'Loyalitas', 'Skor_Nilai','Kinerja'};
insert(conn,'nilai_kinerja',{'NIK', 'Nama', 'Absensi', 'Prestasi',
'Disiplin', 'Loyalitas', 'Skor_Nilai','Kinerja'},data);
msgbox('data berhasil disimpan kedalam
database','pemberitahuan')

```

<code>close(conn);</code>
<b>4. Sourcecode button Laporan</b>
<code>function pushbutton4_Callback(hObject, eventdata, handles)</code> <code>% memanggil fungsi laporan hasil penilaian</code> <code>laporanhasil;</code>
<b>5. Sourcecode button keluar</b>
<code>function pushbutton5_Callback(hObject, eventdata, handles)</code> <code>% keluar dari plikasi penilaian kinerja</code> <code>selection=questdlg(['Anda Yakin Ingin Keluar','?'],...</code> <code>['Keluar' " '...'],...</code> <code>'Ya','tidak','Ya');</code> <code>if strcmp(selection,'tidak')</code> <code>    return</code> <code>end</code> <code>close;</code>

Selanjutnya setelah semua sourcecode dimasukan, Jalankan aplikasi penillaian tersebut dan lakukanlah pengujian sistem secara menyeluruh dan terperinci berkaitan dengan fungsi-fungsi di setiap form dengan menyesuaikan alur proses bisnis. Perhatikan gambar dibawah ini.

Aplikasi\_Utama

File View

APLIKASI PENILAIAN KARYAWAN

NIK

234556

CARI

NAMA

ABSENSI

PRESTASI

DISIPLIN

LOYALITAS

SKOR NILAI

STATUS KINERJA

PROSES

SIMPAN

LAPORAN

KELUAR

	NIK	NAMA	ABSENSI	PRESTASI	DISIPLIN	LOYALITAS	SKOR	KINERJA
1								
2								
3								
4								

**Gambar 4.14** Tampilan Awal Aplikasi Penilaian Kinerja

Pada tampilan awal halaman ini, pengguna harus memasukkan NIK karyawan terlebih dahulu untuk kemudian dilakukan penilaian kinerja, setelah itu pengguna melakukan proses perhitungan dengan menekan tombol proses, setelah proses penilaian selesai, gunakan tombol simpan untuk menyimpan hasil penilaian tersebut

171

## DAFTAR PUSTAKA

- Adhar, D. (2014, September). Sistem Pendukung Keputusan Pengangkatan Jabatan Karyawan pada PT.Ayn dengan Metode Profile Matching. *Jatisi*, 1, 16-28.
- Alamsyah, & Muna, I. H. (2016). Metode Fuzzy Inference System untuk Penilaian Kinerja Pegawai Perpustakaan dan Pustakawan. *Scientific Journal of Informatics*, 88-98.
- Anbarasu, V., Jenitha, A., & Yulit, J. J. (2015). Employee Performance Appraisal System Using Fuzzy Logic. *International Journal Of Innovative Research & Development*, 232-239.
- Anto, A. G., Mustafidah, H., & Suyadi, A. (2015). Sistem Pendukung Keputusan Penilaian Kinerja Karyawan Menggunakan Metode SAW (Simple Additive Weighting) di Universitas Muhammadiyah Purwokerto. *JUITA*, 193 - 200.
- Berutu, E. (2015). Sistem Pendukung Keputusan Pengangkatan Karyawan Tetap Dengan Metode Analytic Hierarchy Process (AHP) Pada PT. Perkebunan Lembah Bhakti Propinsi NAD Kab. Aceh Singkil. *Pelita Informatika Budi Darma*, IX, 96-106.
- Darmastuti, D. (2013). *Implementasi Metode Simple Additive Weighting Dalam Sistem Informasi Lowongan Kerja Berbasis Web Untuk Rekomendasi Pencari Kerja Terbaik*.
- Dewantoro, A. (2013). Sistem Pendukung Keputusan Menentukan Guru Teladan Di SMPN 24 SEMARANG Dengan Menggunakan Metode Simple Additive Weighting.
- Hartati, S. (2010). Analisis Sistem Informasi.
- Haryadi, S., & Rahman, A. A. (2017). Sistem Pendukung Keputusan Penilaian Kinerja Karyawan Menggunakan Metode Weighted Product (WP) Di Pt Infomedia Solusi Humanika Bandung. *JURNAL LPKIA*, 1-6.
- Hermawan, J. (2005). *Membangun Decision Support System*. Yogyakarta: CV. ANDI OFFSET.

- Karmilasari. (2008). *Logika Fuzzy*.
- Kusrini. (2007). *Konsep dan Aplikasi Sistem Pendukung Keputusan*. Yogyakarta: CV. ANDI OFFSET.
- Kusumadewi, S. (2003). *Artificial Intelligence (Teknik & Aplikasinya)*. Yogyakarta: Graha Ilmu.
- Kusumadewi, S. (2006). *Fuzzy Multi-Attribute Decision Making (Fuzzy MADM)*. Yogyakarta: Graha Ilmu.
- Kusumaning Diah R, A. (2013). *Sistem Pendukung Keputusan Rekrutmen Karyawan Produksi Menggunakan Metode WP Pada PT. PLOSS ASIA SEMARANG*.
- Mallu, S. (2015). *Sistem Pendukung Keputusan Penentuan Karyawan Kontrak Menjadi Karyawan Tetap Dengan Metode TOPSIS*.
- MathWorks. (2011). *Creating Graphical User Interfaces*. USA: MATLAB.
- Minartiningtyas, B. A. (2013). *Pengembangan Sistem Pendukung Keputusan*.
- Nurchahyo, S. (2013). *Object Oriented Programming Using C++*. Yogyakarta: ANDI.
- Nurjaya. (2022). *Jurus Kilat Membangun Decision Support System (DSS)*. Tangerang: Pascal Books.
- Prasetya, A. H. (2010). *Cepat Kuasai PHP & MySql*. Yogyakarta: ANDI.
- Rani, S. (2013). In *Modul Pelatihan Pemrograman Matlab*. Yogyakarta: HIMPASIKOM UGM.
- Setiyowati, S. (2013). analisa evaluasi kinerja pegawai untuk promosi. *Seminar nasional aplikasi teknologi informasi*, 9.
- Shaout, A., & Yousif, M. K. (2014). Employee Performance Appraisal System Using Fuzzy Logic. *International Journal of Computer Science & Information Technology (IJCSIT)*, 1-19.

- Sri Kusumadewi. (2006). Fuzzy Multi-Attribute Decision Making (Fuzzy MADM). Yogyakarta: Graha Ilmu.
- Subakti, I. (2003). Sistem Pendukung Keputusan.
- Sudarma S. (2010). *Panduan Belajar MySql Database Server*. Jakarta Selatan: Media Kita.
- Sugiharto, A. (2006). *Pemrograman GUI Dengan MATLAB*. Semarang: Andi Publisher.
- Sugiharto, A. (2006). *Pemrograman GUI Dengan MATLAB*. Yogyakarta: C.V ANDI OFFSET.
- Sugijono. (2015). Penilaian Kinerja Dalam Manajemen Sumber Daya Manusia. *ORBITH Politeknik Negeri Semarang*, 214 – 222.
- Tanaamah, A. R., Advendi, N. G., & Pakereng, I. (2012). Sistem Penilaian Kinerja Pegawai Menggunakan Logika Fuzzy (STUDI KASUS PT. PLN PERSERO APD BALI). *SMATIKA Jurnal*, 7-13.
- Tri Hartini, R. (2014). *Sistem Pendukung Keputusan Pengangkatan karyawan Tetap PT. Rosan Abadi Dengan Metode Satisficing Menggunakan Visual Basic 6.0*.
- Vasanti, G. (2017). Teacher's Performance Appraisal System Using Fuzzy Logic- A Case Study. *International Journal on Recent and Innovation Trends in Computing and Communication*, 273 – 278.
- Wahyudi, S., Suheri, H., & Nurhadian, T. (2015). Implementasi Sistem Keputusan Pengangkatan Karyawan Tetap PT. Imanuel Surya Utama Menggunakan Metode SAW. *Jurnal Sistem Komputer*, 2, 40-48.

## TENTANG PENULIS



**Ahmad**, Lahir di Tangerang pada tahun 1987, Menyelesaikan pendidikan SD,SMP di kota Tangerang Selatan dan SMK di kota Tangerang. Pada tahun 2016 menyelesaikan jenjang pendidikan Strata 1 di Universitas Pamulang, pada tahun 2019 menyelesaikan jenjang pendidikan Strata 2 di STMIK ERESHA. Sejak tahun 2020 sampai saat ini, juga aktif mengajar di salah satu universitas pamulang di tangerang selatan.



**Nurjaya**, Lahir di kota Tangerang pada tahun 1985, Menyelesaikan pendidikan SD, SMP dan SMA di kota Tangerang, pada tahun 2008 menyelesaikan jenjang pendidikan Strata 1, pada tahun 2012 menyelesaikan jenjang pendidikan Strata 2 di STMIK Eresha. Selain aktif sebagai Praktisi dan konsultan IT, sejak tahun 2010 sampai saat ini, juga aktif mengajar di salah satu universitas swasta di Tangerang selatan