BAB II
LANDASAN TEORI

2.1 Pekerjaan

Pekerjaan adalah suatu kegiatan atau suatu tindakan yang menghasilkan sesuatu yang biasanya berupa materi. Pekerjaan ini dapat dikelompokkan menjadi dua, yaitu sebagai berikut :

1. Pekerjaan yang menuntut keahlian dan pendidikan khusus. Contoh : guru, dokter, dan pilot.
2. Pekerjaan yang tidak memerlukan keahlian dan pendidikan khusus. Contoh : kuli panggul dan tukang becak.

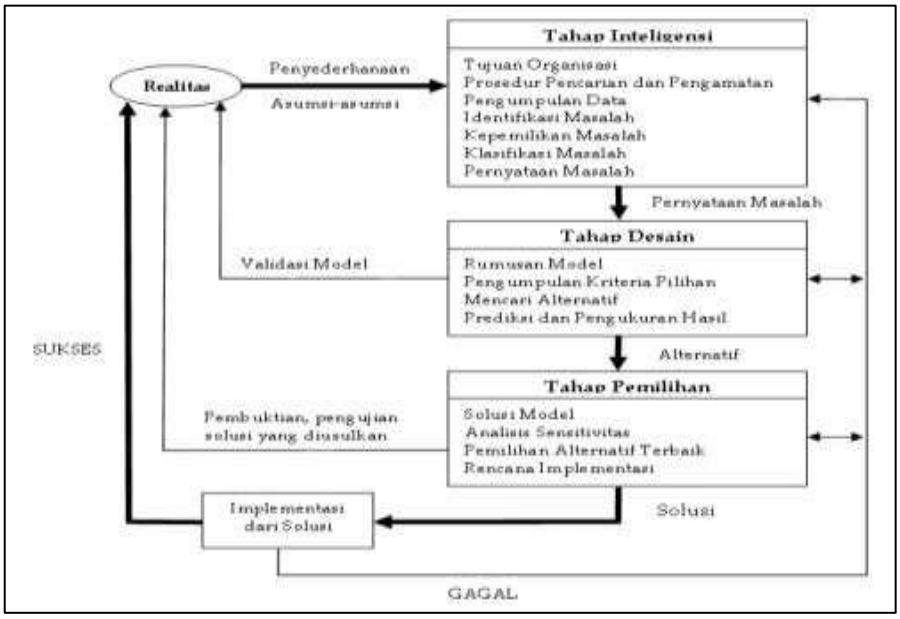
Jadi pada prinsipnya setiap orang dimungkinkan memiliki pekerjaan namun tidak semua pekerjaan itu sama jenisnya karena hal tersebut diukur dari tingkat kesulitan dan pendidikan yang ditempuh oleh orang itu untuk melakukan pekerjaan itu sendiri.

2.2 Konsep Sistem Pendukung Keputusan

(Kusrini, 2007) Konsep sistem pendukung keputusan diperkenalkan pertama kali oleh Michael S. Scoot Morton pada tahun 1970an dengan istilah *Management Decision System* (Sprague, 1982). Pada dasarnya pengambilan keputusan merupakan suatu bentuk pemilihan dari berbagai alternatif tindakan yang dipilih, yang prosesnya melalui mekanisme tertentu dengan harapan akan menghasilkan suatu keputusan yang terbaik.

Proses pendukung keputusan dimulai dengan fase *intelligence*, dimana kenyataan diuji dan masalahnya diidentifikasi. Kemudian fase desain, yaitu suatu model yang menggambarkan bagaimana sistem dibangun, fase ini membuat asumsi sederhana

dengan mengacu pada peraturan-peraturan dan kriteria-kriteria yang sifatnya sudah baku dan menggabungkan antara semua variabel. Selanjutnya model divalidasi dan kriteria-kriteria dikumpulkan untuk suatu evaluasi dari pilihan aksi yang diidentifikasi. Selanjutnya fase pemilihan yang mengandung suatu tujuan penyelesaian dalam model. Fase yang terakhir adalah fase implementasi, dimana akan dilihat tingkat kesuksesan sistem dalam menyelesaikan masalah yang ada (Turban, 1998). Alur proses pengambilan keputusan dapat dilihat pada Gambar 2.1.



Gambar 2.1. Alur Pengambilan Keputusan (Sumber : Turban, 1998).

2.3 Logika *Fuzzy*

(Kusumadewi *et al.*, 2013) Teori logika *fuzzy* diperkenalkan oleh Lotfin A. Zaedah pada tahun 1965. *Fuzzy* secara bahasa diartikan sebagai kabur atau samar-samar. Suatu nilai dapat bernilai benar atau salah secara bersamaan. Dalam *fuzzy* dikenal derajat keanggotaan yang memiliki rentang nilai 0 hingga 1. Berbeda dengan himpunan tegas yang memiliki nilai 1 atau 0 (ya atau tidak). Logika *fuzzy* adalah suatu cara yang tepat untuk memetakan suatu ruang input kedalam suatu ruang output, mempunyai nilai kontinyu. Oleh sebab itu, sesuatu dapat dikatakan sebagian benar dan sebagian salah pada waktu yang sama.

Logika *fuzzy* memungkinkan nilai keanggotaan antara 0 dan 1, tingkat keabuan, dan juga hitam dan putih, dan dalam bentuk linguistik, konsep tidak pasti seperti "sedikit", "lumayan", dan "sangat" (Zadeh, 1965). Kelebihan dari teori logika *fuzzy* adalah kemampuan dalam proses penalaran secara bahasa (*linguistic reasoning*) sehingga dalam perancangannya tidak memerlukan persamaan matematik dari objek yang akan dikendalikan. Dasar logika *fuzzy* adalah teori himpunan *fuzzy*, dimana di dalam himpunan *fuzzy* komponen utama yang sangat berpengaruh adalah fungsi keanggotaan. Fungsi keanggotaan merepresentasikan derajat kedekatan suatu obyek terhadap atribut tertentu (Ross, 2005).

Terdapat beberapa hal yang perlu diketahui dalam memahami sistem *fuzzy*, yaitu: Variabel *fuzzy* yang merupakan variabel yang hendak dibahas dalam sistem *fuzzy*. Himpunan *fuzzy* yang merupakan suatu grup yang mewakili suatu kondisi atau keadaan tertentu dalam suatu variabel *fuzzy*. Semesta pembicaraan yang merupakan keseluruhan nilai yang diperbolehkan untuk dioperasikan dalam suatu varibel *fuzzy* dan merupakan himpunan bilangan real yang senantiasa bertambah secara monoton dari kiri ke kanan serta adakalnya nilai semesta pembicaraan ini tidak dibatasi batas atasnya. Domain himpunan *fuzzy* yang merupakan keseluruhan nilai yang diizinkan dalam semesta pembicaraan dan boleh dioperasikan dalam suatu himpunan *fuzzy*. Selain itu di dalam sistem *fuzzy* perlu dipahami jenis atribut dari setiap kriteria, yaitu *benefit* dan *cost*. Atribut *benefit* digunakan pada kriteria yang memiliki nilai kecocokan dengan ketentuan nilai terbesar adalah yang terbaik, sehingga kriterianya disebut sebagai kriteria keuntungan. Sedangkan atribut *cost* merupakan nilai yang diberikan pada setiap altrnatif di setiap kriteria dengan nilai kecocokan nilai terkecil adalah yang terbaik dan juga digunakan pada atribut yang berkaitan dengan biaya, sehingga kriterianya disebut sebagai kriteria biaya.

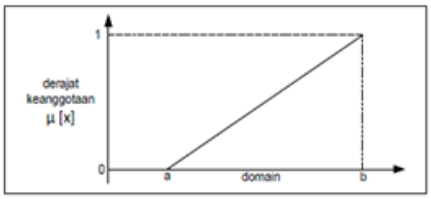
2.3.1 Fungsi Keanggotaan

Fungsi keanggotaan (*membership function*) adalah suatu kurva yang menunjukkan pemetaan titik-titik input data ke dalam nilai keangotaannya (sering disebut juga dengan derajat keanggotaan). Salah satu cara yang dapat digunakan untuk mendapatkan nilai keanggotaan adalah dengan melalui pendekatan fungsi.

Terdapat beberapa fungsi yang bisa digunakan (Kusumadewi, 2013), diantaranya :

1. Representasi Linear

Pada representasi linear, pemetaan input ke derajat keanggotaannya digambarkan sebagai suatu garis lurus. Ada 2 keadaan himpunan *fuzzy* yang linear. Pertama, kenaikan himpunan dimulai pada nilai dominan yang memiliki derajat keanggotaan nol [0] bergerak ke kanan menuju ke nilai dominan yang memiliki derajat keanggotaan lebih tinggi terlihat pada Gambar 2.2.

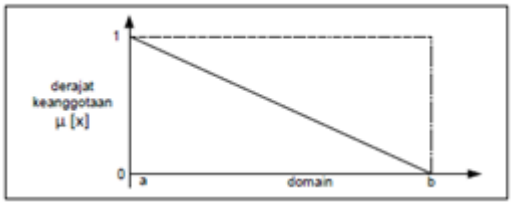


Gambar 2.2**.** Representasi Linear Naik.

Fungsi keanggotaan :

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

Kedua, garis lurus dimulai dari nilai dominan dengan derajat keanggotaan tertinggi pada sisi kiri, kemudian bergerak menurun ke nilai dominan yang memiliki derajat keanggotaan lebih rendah terlihat pada Gambar 2.3.



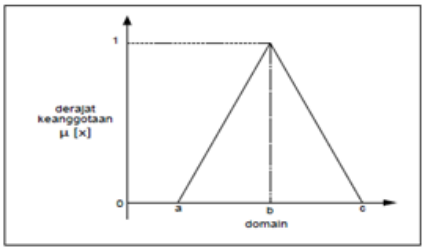
Gambar 2.3. Representasi Linear Turun.

Fungsi keanggotaan :

$$\mu\ x = \begin{cases} \dfrac{b-x}{b-a}; & a \le x \le b \\ 0; & x \ge b \end{cases}$$

2. Representasi Kurva Segitiga

Kurva segitiga merupakan gabungan antara 2 garis (linear) seperti terlihat pada Gambar 2.4.



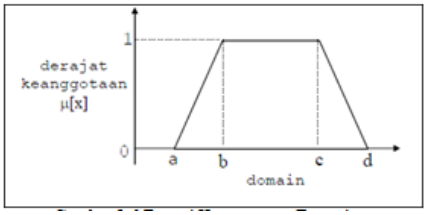
Gambar 2.4. Kurva Segitiga.

Fungsi keanggotaan :

$$\mu\ x = \begin{cases} 0; & x \le a\ \ atau\ \ x \ge c \\ \dfrac{x-a}{b-a}; & a \le x \le b \\ \dfrac{b-x}{c-b}; & b \le x \le c \end{cases}$$

3. Representasi Kurva Trapesium

Kurva trapesium pada dasarnya seperti bentuk pada kurva segitiga, namun ada beberapa titik yang memiliki nilai keanggotaan 1 seperti terlihat pada Gambar 2.5.



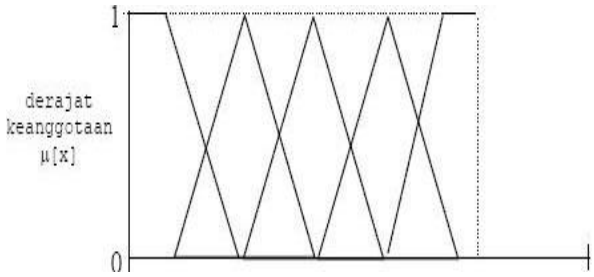
Gambar 2.5**.** Kurva Trapesium.

Fungsi keanggotaan :

$$\mu\ x\ = \begin{cases} 0; & x \leq a\ atau\ x \geq d \\ \dfrac{x-a}{b-a}; & a \leq x \leq b \\ 1; & b \leq x \leq c \\ \dfrac{d-x}{d-c}; & x \geq d \end{cases}$$

4. Representasi Kurva Berbentuk Bahu

Pada kurva berbentuk bahu, daerah yang terletak di tengah-tengah suatu variabel yang direpresentasikan dalam bentuk segitiga pada sisi kanan dan kirinya akan naik dan turun seperti terlihat pada Gambar 2.6.



Gambar 2.6**.** Kurva Bentuk Bahu.

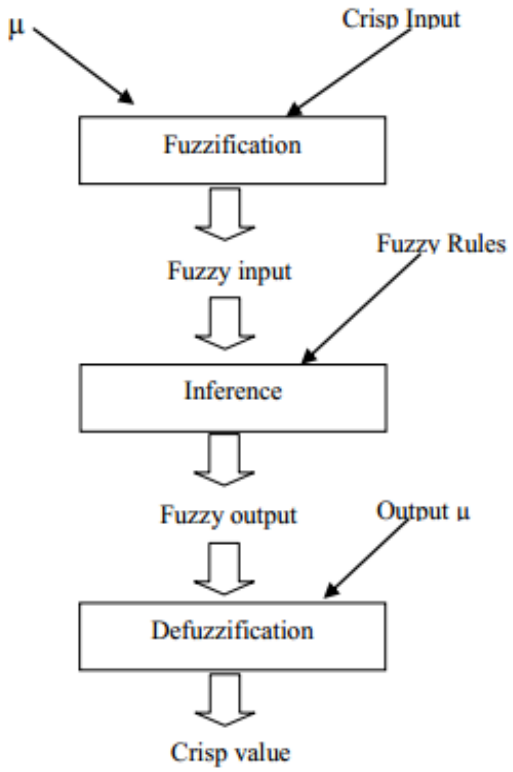
2.4 *Fuzzy inference system* (FIS)

Motivasi utama teori fuzzy logic adalah memetakan sebuah ruang input ke dalam ruang output dengan menggunakan IF-THEN rules. Pemetaan dilakukan dalam suatu Sistem Inferensi Fuzzy (Fuzzy Inference System/FIS) disebut juga fuzzy inference engine adalah sistem yang dapat mengevaluasi semua rule secara simultan untuk menghasilkan kesimpulan dan urutan rule bisa sembarang (Naba, 2009). Oleh karennya, semua aturan atau rule harus

didefinisikan lebih dahulu sebelum membangun sebuah FIS yang akan digunakan untuk menginterprestasikan sebuah rule tersebut.

Terdapat beberapa jenis FIS yang dikenal yaitu Mamdani, Sugeno dan Mamdani (Naba, 2009). Suatu sistem berbasis aturan fuzzy yang lengkap terdiri dari tiga komponen utama : *fuzzification*, *inference*, dan defuzzification seperti pada diagram blok sistem berbasis aturan pada Gambar 2.7.

Fuzzification mengubah masukan-masukan yang nilai kebenarannya bersifat pasti (crisp input) ke dalam bentuk fuzzy input, yang berupa nilai linguistic yang semantiknya ditentukan berdasarkan fungsi keanggotaan (μ) tertentu. Inference melakukan penalaran menggunakan fuzzy input dan fuzzy rules yang telah ditentukan sehingga menghasilkan fuzzy output. Sedangkan defuzzification mengubah fuzzy output menjadi crisp value berdasarkan fungsi keanggotaan (μ) yang telah ditentukan.



Gambar 2.7 Diagram blok sistem berbasis aturan fuzzy (Suyanto, 2007)

2.5 Metode *Fuzzy* MAMDANI

Metode Mamdani sering dikenal sebagai Metode Max-Min. Metode ini diperkenalkan oleh Ebrahim Mamdani pada tahun 1975. Untuk mendapatkan *output*, diperlukan 4 tahapan:

1. **Pembentukan himpunan fuzzy**

   Pada Metode Mamdani, baik variabel *input* maupun variable *output* dibagi menjadi satu atau lebih himpunan fuzzy.

2. **Aplikasi fungsi implikasi**

   Pada Metode Mamdani, fungsi implikai yang digunakan adalah Min.

3. **Komposisi Aturan**

   Ada 3 metode yang digunakan dalam melakukan inferensi sistem fuzzy,yaitu: max, additive dan probabilistik OR (probor).

   a. Metode Max (Maximum)

      Pada metode ini, solusi himpunan fuzzy diperbolehkan dengan cara menggambil nilai maksimum aturan, kemudian menggunakannya untuk memodifikasi daerah fuzzy, dan mengaplikasikannya ke *output* dengan menggunakan operator OR (*union*). Jika semua posisi telah dievaluasi, maka *output* akan berisi suatu himpunan fuzzy yang mereflesikan kontibusi dari tiap-tiap proposisi. Secara umum dapat dituliskan:

$$\mu_{sf}\ x_i = \max(\mu_{sf}\ x_i\ , \mu_{kf}\ x_i\ )$$

      Dengan:

      $\mu_{sf}\ x_i$ = nilai keanggotaan solusi fuzzy sampai aturan ke-I;

      $\mu_{kf}\ x_i$ = nilai keanggotaan konsekuen fuzzy aturan ke-I;

      Misalkan ada 3 aturan (proposisi) sebagai berikut:

      [R1] IF Biaya Produksi Rendah And Permintaan NAIK
      THEN Produksi Barang BERTAMBAH;

      [R2] IF Biaya Produksi STANDAR
      THEN Produksi Barang NORMAL;

      [R3] IF Biaya Produksi TINGGI And Permintaan TURUN
      THEN Produksi Barang BERKURANG;

Proses inferensi dengan menggunakan metode Max dalam melakukan komposisi aturan. Apabila digunakan fungsi fungsi implikasi MIN, maka metode komposisi ini sering disebut dengan nama MAX-MIN atau MIN-MAX atau MAMDANI.

b. Metode *Additive* (Sum)

Pada metode ini, solusi himpunan fuzzy diperoleh dengan cara melakukan *bounded-sum* terhadap semua *output* daerah fuzzy.

Secara umum dituliskan:

$$\mu_{sf}\ x_i\ = \min\ \ 1, \mu_{sf}\ x_i\ +\ \mu_{kf}\ x_i$$

Dengan :

$\mu_{sf}\ x_i$ = nilai keanggotaan solusi fuzzy sampai aturan ke-I;

$\mu_{kf}\ x_i$ = nilai keanggotaan konsekuen fuzzy aturan ke-I;

c. Metode Probabilistik OR (probor)

Pada metode ini, solusi himpunan fuzzy diperoleh dengan cara melakukan *product* terhadap semua *output* daerah fuzzy. Secara umum dituliskan:

$$\mu_{sf}\ x_i\ = (\mu_{sf}\ x_i\ +\ \mu_{kf}\ x_i\ -\ \mu_{sf}\ x_i\ *\ \mu_{kf}\ x_i\ )$$

Dengan :

$\mu_{sf}\ x_i$ = nilai keanggotaan solusi fuzzy sampai aturan ke-I;

$\mu_{kf}\ x_i$ = nilai keanggotaan konsekuen fuzzy aturan ke-I;

4. Penegasan (*defuzzy*)

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

Ada beberapa metode defuzzifikasi pada komposisi aturan MAMDANI, antara lain:

a. Metode *Centroid* (*Composite Moment*)

Pada metode ini, solusi *crisp* diperoleh dengan cara mengambil titik pusat (z*) daerah fuzzy. Secara umum dirumuskan:

$Z^* = \frac{{}_z \; z\mu \; z \; dz}{{}_z \; \mu \; z \; dz}$ untuk variable kontinu, atau

$Z^* = \frac{{}_{j=1}^{n} z \;\; {}_j \mu(z_j)}{{}_{j=1}^{n} z \; \mu(z_j)}$ untuk variable disket.

b. Metode Bisektor

Pada metode ini, solusi *crisp* diperoleh dengan cara mengambil nilai rata-rata domain fuzzy yang memiliki nilai keanggotaan setengah dari jumlah total nilai kenggotaan pada daerah fuzzy.

c. Metode *Mean of Maximum* (MOM)

Pada metode ini, solusi *crisp* diperoleh dengan cara mengambil nilai rata-rata domain yang memiliki nilai keanggotaan maksimum.

d. Metode *Largest of Maximum* (LOM)

Pada metode ini, solusi *crisp* diperoleh dengan cara mengambil nilai terbesar dari domain yang memiliki nilai kenanggotaan maksimum.

e. Metode *Samllestof Maximum* (SOM)

Pada metode ini, solusi *crisp* diperoleh dengan cara mengambil nilai terkecil dari domain yang memiliki nilai keanggotaan maksimum.

2.6 *Website*

(YM.Khusuma Ardhama, 2012) *Website* adalah kumpulan dari beberapa halaman *web* dimana informasi dalam bentuk teks, gambar, suara, dan lain-lain yang dipersentasikan dalam bentuk *hypertext* dan dapat diakses oleh perangkat lunak yang disebut dengan *browser*. Informasi pada sebuah *website* pada umumnya ditulis dalam format HTML. Informasi lainya disajikan dalam bentuk grafis (dalam format GIF, JPG, PNG, dll), suara (dalam format AU, WAV, dll), dan objek multimedia lainya (seperti MIDI, *Shockwave Quicktime Movie, 3D World*, dll*).*

Website merupakan fasilitas internet yang menghubungkan dokumen dalam ruang lingkup lokal maupun jarak jauh. Dokumen pada *website* disebut dengan

webpage dan *link* dalam *website* memungkinkan pengguna bisa berpindah dari satu *page* ke *page* lain (*hypertext)*, baik diantara *page* yang disimpan dalam *server* yang sama maupun *server* diseluruh dunia. *Pages* diakses dan dibaca melalui *browser* seperti *Netscape Navigator* atau *Internet Exploler* berbagai aplikasi *browser* lainnya.

2.7 HTML

(Raharjo, H and RK, 2010) HTML (*HyperText Markup Language*) merupakan *file* teks yang ditulis menggunakan aturan-aturan kode tertentu untuk kemudian disajikan ke *user* melalui suatu aplikasi *web browser*. Setiap informasi yang tampil di *web* selalu dibuat menggunakan kode HTML. Oleh karena itu, dokumen HTML sering disebut juga sebagai *web page* (halaman *web*).

Secara umum, dokumen HTML terbagi atas dua bagian, yaitu bagian ***header*** (kepala) dan ***body*** (badan). Bagian *header* diawali dengan *tag* <*head*> dan diakhiri dengan *tag* <*/head*>. Sedangkan bagian *body* diawali dengan *tag* <*body*> dan ditutup dengan *tag* <*/body*>. Kedua bagian tersebut diapit oleh *tag* <*html*> dan <*/html*>.

2.8 PHP

(YM.Khusuma Ardhama, 2012) *Hypertext Preprocessor* atau sering disebut PHP merupakan bahasa permrograman bebasis *server-side* yang dapat melakukan parsing *script* php menjadi menjadi *script web* sehingga dari sisi *client* menghasilkan suatu tampilan yang menarik. PHP merupkan pengembangan dari FI atau *Form Interface* yang dibuat oleh Rasmus Lerdoff pada tahun 1995.

Berbeda dengan HTML, kode PHP tidak diberikan ssecara langsung oleh server ketika ada permintaan atau request dari sisi client namun dengan cara pemrosesan dari sisi server. Kode PHP disisipkan pada kode HTML. Perbedaan dari kode (*script*) HTML dan PHP yaitu setiap kode PHP ditulis selalu diberi *tag* pembuka yaitu <?php dan pada akhir kode PHP diberi *tag* penutup yaitu ?> .

2.9 DFD (Data *Flow* Diagram)

(S and Shalahuddin, 2016) *Data Flow Diagram* adalah representasi grafik yang menggambarkan aliran informasi dan tranformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*output*). DFD dapat digunakan untuk merepresentasikan sebuah sistem atau perangkat lunak pada beberapa level abstraksi. DFD dapat dibagi menjadi beberapa level yang lebih detail untuk merepsentasikan aliran informasi atau fungsi yang lebih detail. DFD menyediakan mekanisme untuk pemodelan fungsional ataupun pemodelan aliran informasi. Notasi-notasi pada DFD adalah seperti pada Tabel 2.1.

Tabel 2.1 Notasi Data *Flow* Diagram (DFD).

Notasi	Keterangan
	Proses atau fungsi; pada pemodelan perangkat lunak pada pemograman terstruktur, maka pemodelan notasi inilah yang harus menjadi prosedur didalam kode program.
	File atau basis data atau penyimpanan (*storage*) pada pemodelan perangkat lunak,, maka pemodelan notasi inilah yang harusnya dibuat mejadi table-tabel basis data yang dibutuhkan, table-tabel ini juga harus sesuai dengan perancangan table-tabel pada basis data (ERD).
	Entitas luar (external entity) atau masukan atau keluaran atau orang-oarang yang memakai atau berinteraksi dengan perangkat luanak yang dmodelkan atau system lain yang terkait dengan aliran data dari system yang dimodelkan.

2.10 ERD (*Entity Relationship Diagram*)

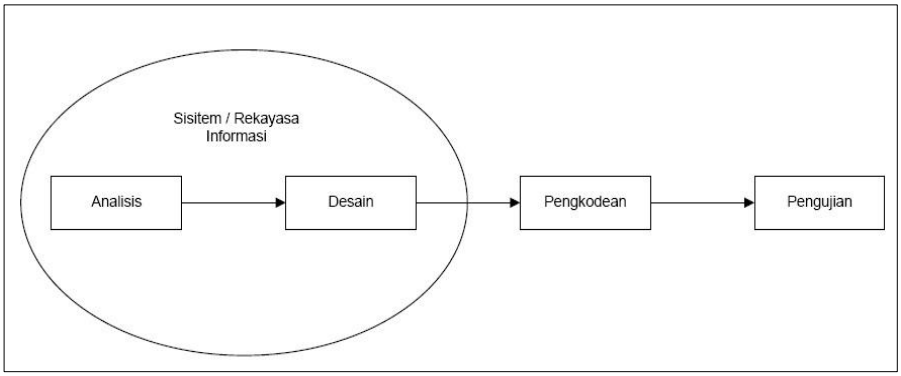
(S and Shalahuddin, 2016) Pemodelan awal basis data yang paling banyak digunakan adalah *Entity Relationship Diagram* (ERD). ERD dikembangkan berdasarkan teori himpunan dalam bidang matematika. ERD digunakan untuk pemodelan basis data relasional. Notasi-notasi pada ERD adalah seperti pada Tabel 2.2.

Tabel 2.2 Notasi Pada Model ERD.

Simbol	Keterangan
Entitas /entity Nama_entit	Entitas/entity merupakan data inti yang akan dismpan; bakal table pada basis data
Atribut Nama_atribut	Atribut/field;atau kolom data yang butuh disimpan dalam suatu entitas
Atribut Kunci Primer Nama_kunci_ .	*Field* atau kolom digunakan sebagai kunci akses recod yang diinginkan; biasanya berupa id
Atribut Multi nilai Nama_atribut	*Field* atau kolom data yang butuh disimpan dalam suatau entitas yang dapat memiliki nilai lebih dari satu.
Relasi Nama_rel asi	Relasi yang menghubungkan antar aentitas; biasanya diawali dengan kata kerja.

2.11 Metode Perancangan Sistem

Metode yang digunakan dalam pembuatan sistem pendukung keputusan pemilihan asisten laboratorium ini dengan menggunakan model *waterfall*. Model *waterfall* dibagi menjadi 5 tahapan (S and Shalahuddin, 2016),yaitu:



Gamabr 2.8 Ilustrasi Model *Waterfall* Sumber (S and Shalahuddin, 2016).

2.11.1 Analisis

Pada tahap ini yang dilakukan adalah menganalisis kebutuhan sistem. Proses pengumpulan kebutuhan dilakukan untuk menspesifikasian kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*.

2.11.2 Desain

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya.

2.11.3 Pengkodean

Hasil dari tahap ini adalah translasi desain ke dalam bentuk program perangkat lunak yang didapatkan dari pembuatan program di komputer.

2.11.4 Pengujian

Tahap ini berfokus pada perangkat lunak secara *logic* dan fungsional. Selain itu, memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan diinginkan.

2.11.5 Pemeliharaan

Sebuah perangkat lunak memungkinkan mengalami perubahan ketika sudah dikirimkan ke *user*. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru.

2.12 Penelitian Terdahulu

Tabel 2. 3 adalah beberapa penelitian yang berkaitan dengan penerapan metode *fuzzy* MAMDANI.

Tabel 2.3 Penelitian Terdahulu.

No.	Nama, Tahun	Nama/Judul Penelitian	Fokus dan Tujuan Penelitian
1	(Yulmaini, 2015)	Penggunaan Metode Fuzzy Inference system (FIS) Mamdani Dalam Pemilihan Peminatan Mahasiswa untuk Tugas Akhir	Penelitiam ini bertujuan untuk menghasilkan suatu sistem fuzzymetode FIS-Mamdani sebagai alternatif penyelesaian masalah dalam pemilihan peminatan untuk tugas akhir. Metode yang digunakan dalam penelitian ini adalah FIS-Mamdani dengan 12 variable input, dan 37 aturan fuzzy, dan 3 peminatan tugas akhir. Hasil

			penelitian ini yaitu sistem fuzzy dengan menggunakan metode FIS-Mamdani yang dapat menyelesaikan masalah dalam penentuan peminatan tugas akhir mahasiswa.
2	(Yulmaini and Indriyati, 2016)	The Design Of The Fuzzy Systems In Selecting The Students' Concentration In Terms Of Determining The Topic Of The Thesis	Penelitian ini bertujuan untuk menghasilkan perancangan sistem fuzzy yang dapat menentukan konsentrasi dan topik tesis dengan menggunakan logika fuzzy. Metode yang digunakan dalam penelitian ini yaitu metode sistem cerdas yang disebut logika fuzzy, digunakan karena dapat mentolerir nilai ketidakpastian dan didasarkan pada penalaran yang menggunakan pendekatan ketidakpastian yang menggabungkan variable numerik, variable linguistik, dan aturan.
3	(Sumitre and Kurniawan, 2014)	Rancangan Bangun Sistem Pendukung Keputusan Seleksi Penerimaan Tenaga Pengajar dengan	Penelitian ini bertujuan untuk menghasilkan sistem yang dapat memberikan alternatif penyelesaian masalah dalam penerimaan tenaga pengajar.

		Metode Fuzzy Inference System (FIS) Mamdani	Dalam Proses pengambilan keputusan penerimaan tenaga pengajar digunakan metode fuzzy mamdani yang sering dikenal sebagai memtode max-min. Variabel input yang digunakan pada penelitian ini yaitu variabel IPK, pengalaman, wawancara, dan ketertarikan sedangkan variabel outputnya adalah nilai akhir.
4	(Darmastuti, 2013)	Implementasi Metode Simple Additive Weighting (SAW) dalam Sistem Informasi Lowong Kerja Berbasis Web untuk Rekomendasi Pencari Kerja Terbaik.	Penelitian ini bertujuan untuk menghasilkan suatu informasi sistem web pekerjaan berdasarkan yang akan mempertemukan pencari kerja dan penyedia kerja dan penyedia kerja. Metode yang digunakan dalam penelitian ini yaitu metode simple additive weighting (SAW). Kriteria yang ditetapkan dalam penelitian ini yaitu nilai IPK minimal, tinggi badan minimal, pendidikan terakhir minimal, usia maksimal, pengalaman kerja yang dibutuhkan, nilai akreditasi universitas, latar belakang program studi yang

			dibutuhkan, kondisi mata pencari kerja, kemampuan berbahasa asing, serta nilai Ujian Nasional (UN) SMA.
5	(Sutisna and Basjaruddin, 2015)	Sistem Pendukung Keputusan Pemilihan Pekerjaan Menggunakan Metode Fuzzy Mamdani Studi Kasus: AMIK BSI Tasikmalaya	Penelitian ini bertujuan untuk menghasilkan sistem pendukung keputusan untuk pemilihan pekerjaan. Penalaran logika fuzzy mamdami digunakan untuk ,e,proses data masukan yang selanjutnya digunakan untuk menghasilkan keluaran berupa rekomendasi pekerjaan apa yang tepat bagi seseorang.