

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Asisten Laboratorium**

Asisten laboratorium merupakan mahasiswa yang berkeinginan menjadi asisten praktikum mata kuliah tertentu. Salah satu tugas asisten praktikum yakni memberi bimbingan dan membantu praktikan dalam pelaksanaan praktikum di laboratorium.

##### **2.1.1 Penerimaan Asisten Laboratorium**

Penerimaan calon asisten laboratorium merupakan kegiatan rutin yang dilakukan setiap tahun ajaran baru (satu tahun sekali). Penerimaan asisten laboratorium dilakukan dengan cara melakukan seleksi terhadap calon asisten sesuai dengan kriteria yang telah ditetapkan, hal ini bertujuan untuk mengoptimalkan proses pembelajaran yang ada. Dalam penerimaan asisten laboratorium terdapat instruksi kerja yang digunakan sebagai acuan bagi pihak laboratorium dalam melakukan proses penerimaan asisten baru yang melibatkan seluruh civitas akademika IBI Darmajaya.

Dalam instruksi kerja ini terkait beberapa pengertian, antara lain:

1. Koordinator Laboratorium Komputer adalah penanggung jawab segala kegiatan yang berkaitan dengan laboratorium
2. Administrator laboratorium adalah penanggung jawab dalam pembagian jadwal asistensi, mengelola administrasi yang berkaitan dengan laboratorium komputer, termasuk mengelola pendaftaran calon asisten laboratorium
3. Laboran adalah penanggung jawab sekaligus pelaksana segala kegiatan yang berkaitan dengan laboratorium
4. Asisten laboratorium adalah pelaksana segala kegiatan yang berkaitan dengan proses pembelajaran

5. Calon asisten merupakan mahasiswa yang berkompeten dan berkeinginan menjadi asisten laboratorium.

Selain instruksi kerja yang berkaitan dengan laboratorium komputer, dalam proses penyeleksian calon asisten terdapat beberapa kualifikasi/kriteria yang harus dimiliki oleh pelamar. Kriteria-kriteria tersebut yang digunakan dalam seleksi pemilihan calon asisten adalah :

#### 2.1.1.1 Semester

Untuk menjadi asisten, salah satu prasyarat dalam seleksi adalah dilihat dari semester calon asisten. Minimal semester adalah 3 dan maksimal semester 8. Pada kriteria semester, calon asisten dianggap memumpuni karena telah mengambil mata kuliah praktikum. Selain itu, juga dianggap memiliki kemampuan dan memahami mata kuliah di kelas praktikum tersebut.

#### 2.1.1.2 IPK

IPK merupakan salah satu kriteria yang juga penting dalam seleksi asisten. IPK adalah standar akademik yang digunakan sebagai landasan atau patokan calon asisten karena IPK berkaitan dengan nilai mata kuliah yang telah diambil. Standar minimal untuk menjadi asisten adalah dengan IPK 3,00.

#### 2.1.1.3 Nilai Mata Kuliah Praktikum

Nilai mata kuliah praktikum adalah nilai yang diperoleh calon asisten di mata kuliah yang terdapat praktikumnya di laboratorium komputer. Nilai praktikum merupakan hasil evaluasi calon asisten selama mengampu mata kuliah praktik tersebut. Dalam seleksi asisten, standar untuk penilaian nilai praktikum adalah disesuaikan dengan standar nilai akademik di IBI Darmajaya.

#### 2.1.1.4 Tes Tertulis

Tes tertulis merupakan bentuk dari alat ukur yang digunakan untuk mengetahui kemampuan calon asisten dalam pemahaman secara teoritis. Dalam tes tertulis, calon asisten diberikan sejumlah pertanyaan secara tertulis yang berkaitan dengan

materi praktikum. Standar yang digunakan laboratorium dalam tes tertulis adalah 70.

#### 2.1.1.5 Tes Kompetensi

Tes kompetensi merupakan bentuk tes praktik langsung di laboratorium komputer untuk menguji kemampuan yang dimiliki calon asisten dalam beberapa bentuk uji *software* yang berkaitan dengan materi praktikum. Tes kompetensi sangat penting dalam pemilihan asisten karena berkaitan dengan tugas asisten yaitu bertanggung jawab dalam membantu dosen dalam menyampaikan materi secara praktik di laboratorium. Sehingga calon asisten diuntut dapat menguasai mata kuliah praktikum. Standar yang digunakan laboratorium dalam tes ini adalah sama dengan tes tertulis, yaitu sebesar 70.

#### 2.1.1.6 Tes Wawancara

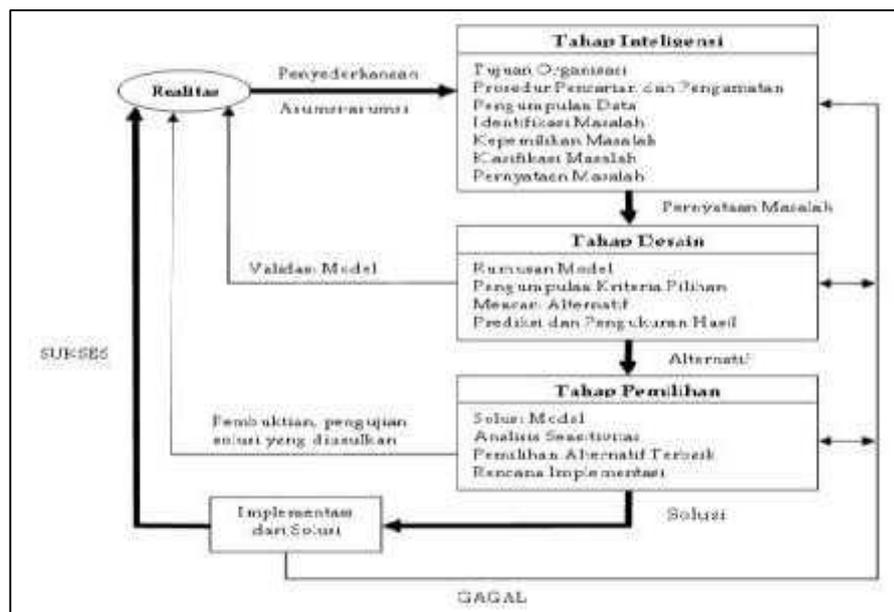
Wawancara merupakan tahap terakhir dari proses seleksi asisten, dimana calon asisten diberikan pernyataan langsung oleh tim penyeleksi guna mengetahui komitmen dan tanggung jawab apabila menjadi asisten laboratorium komputer. Standar nilainya pun sama, yaitu 70.

## 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 variabel *fuzzy* dan merupakan himpunan bilangan real yang senantiasa bertambah secara monoton dari kiri ke kanan serta adakalanya 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 alternatif 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.

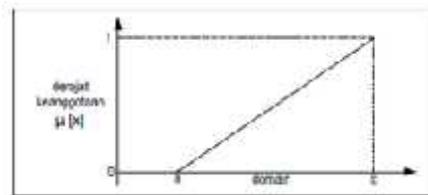
### **2.3.1 Fungsi Keanggotaan**

Fungsi keanggotaan (*membership function*) adalah suatu kurva yang menunjukkan pemetaan titik-titik input data ke dalam nilai keanggotaannya (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.

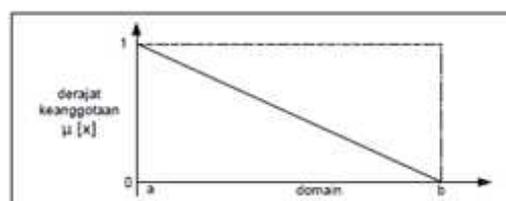


Gambar 2.2. Representasi Linear Naik.

Fungsi keanggotaan :

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

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.



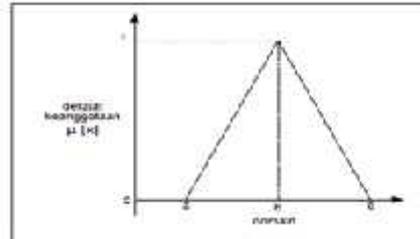
Gambar 2.3. Representasi Linear Turun.

Fungsi keanggotaan :

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

## 2. Representasi Kurva Segitiga

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



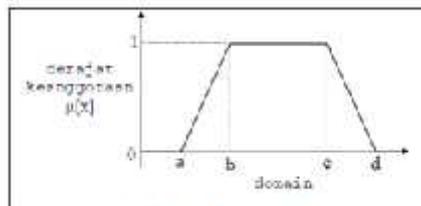
Gambar 2.4. Kurva Segitiga.

Fungsi keanggotaan :

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

## 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.



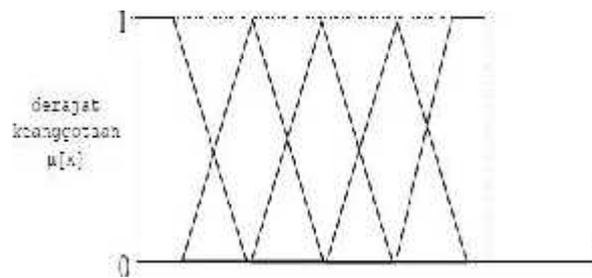
Gambar 2.5. Kurva Trapesium.

Fungsi keanggotaan :

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

#### 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 Multiple Attribute Decision Making (FMADM)

*Fuzzy Multiple Attribute Decision Making* (FMADM) adalah suatu metode yang digunakan untuk mencari alternatif optimal dari sejumlah alternatif dengan kriteria tertentu. Inti dari FMADM adalah menentukan nilai bobot untuk setiap atribut, kemudian dilanjutkan dengan proses perankingan yang akan menyeleksi alternatif yang sudah diberikan.

Pada dasarnya terdapat 3 pendekatan untuk mencari nilai bobot atribut, yaitu pendekatan subyektif, pendekatan obyektif dan pendekatan integrasi antara subyektif & obyektif. Masing-masing pendekatan memiliki kelebihan dan kelemahan. Pada pendekatan subyektif, nilai bobot ditentukan berdasarkan subyektifitas dari para pengambil keputusan, sehingga beberapa faktor dalam proses perankingan alternatif bisa ditentukan secara bebas. Sedangkan pada pendekatan obyektif, nilai bobot dihitung secara matematis sehingga mengabaikan subyektifitas dari pengambil keputusan. Selain itu berdasarkan tipe data yang digunakan pada setiap kinerja alternatif-alternatifnya, FMADM dapat dibagi menjadi 3 kelompok, yaitu: semua data yang digunakan adalah data *fuzzy*, semua data yang digunakan adalah *crisp*, dan data yang digunakan merupakan campuran antara data *fuzzy* dan *crisp*.

Ada beberapa metode yang dapat digunakan untuk menyelesaikan masalah FMADM seperti *Simple Additive Weighting Method (SAW)*, *Weighted Product (WP)*, ELECTRE, *Technique for Order Preference by Similarity to Ideal Solution (TOPSIS)*, dan *Analytic Hierarchy Process (AHP)* untuk melakukan proses perengkingan dimana terlebih dahulu dilakukan konversi dari data *fuzzy* ke data *crisp*.

## 2.5 Metode *Fuzzy* TOPSIS

(Kusumadewi et al. 2013) *Fuzzy* TOPSIS merupakan metode yang dapat membantu dalam evaluasi yang sistematis dari beberapa alternatif pada kriteria dimana setiap kriteria memiliki nilai *fuzzy*. Konsep metode *fuzzy* TOPSIS sederhana dan mudah dipahami, komputasinya efisien, dan memiliki kemampuan untuk mengukur kinerja relatif dari alternatif-alternatif keputusan dalam bentuk matematis sederhana.

Metode *fuzzy* TOPSIS didasarkan pada konsep, dimana alternatif terpilih tidak hanya memiliki jarak terpendek dari solusi ideal positif, namun juga memiliki jarak terpanjang dari solusi ideal negatif. Sehingga alternatif yang dipilih memiliki kedekatan dengan solusi ideal positif dan jauh dari solusi ideal negatif. Solusi ideal terbentuk jika sebagai komposit dari nilai kinerja terbaik ditampilkan oleh setiap alternatif untuk setiap atribut dan solusi ideal negatif adalah gabungan dari nilai kinerja terburuk. Jarak ke masing-masing kutub kinerja diukur dalam pengertian *Euclidean*, dengan bobot opsional dari setiap atribut (Kahraman C, 2008).

Prosedur dari metode *fuzzy* TOPSIS mengikuti langkah-langkah sebagai berikut (Kusumadewi et al. 2013):

1. Menentukan kriteria yang terkait dengan studi kasus
2. Menentukan bobot dan tingkat kepentingan dari setiap kriteria
3. Menentukan derajat keanggotaan setiap kriteria pada setiap alternatif
4. Melakukan konversi bilangan *fuzzy* matriks keputusan kedalam bentuk bilangan *crisp*
5. Menentukan matriks keputusan yang ternormalisasi (**R**)

TOPSIS membutuhkan rating kinerja tiap calon karyawan pada setiap kriteria atau subkriteria yang ternormalisasi. Matriks ternormalisasi terbentuk dari persamaan :

$$R = \frac{X}{\sqrt{\sum_{i=1}^n x_i^2}} \quad p \quad (1)$$

6. Menghitung matriks keputusan yang ternormalisasi terbobot ( $Y$ )

Untuk menghitung matriks ternormalisasi terbobot  $Y_{ij}$ , maka harus ditentukan terlebih dahulu nilai bobot  $W$  yang merepresentasikan preferensi *absolute* dari pengambil keputusan. Nilai bobot preferensi menunjukkan tingkat kepentingan relatif setiap kriteria atau subkriteria pada persamaan :

$$W = \{W_1, W_2, \dots, W_n\} \quad Y = W \cdot R \quad p \quad (2)$$

7. Menghitung matriks solusi ideal positif dan matriks solusi ideal negatif ( $A^+$  dan  $A^-$ ).

Solusi ideal positif dan solusi ideal negatif dapat ditentukan berdasarkan rating bobot ternormalisasi. Perlu diperhatikan syarat untuk menghitung nilai solusi ideal dengan terlebih dahulu menentukan apakah bersifat keuntungan (*benefit*) atau bersifat biaya (*cost*), sehingga :

$$A^+ = \{Y_1^+, Y_2^+, \dots, Y_n^+\} \text{ persamaan (3)}$$

$$A^- = \{Y_1^-, Y_2^-, \dots, Y_n^-\} \text{ persamaan (4)}$$

Dengan syarat :

$$y_j^+ = \begin{cases} \max y_i & ; \text{ jika } j \text{ adalah atribut } b \\ \min y_i & ; \text{ jika } j \text{ adalah atribut } c_i \end{cases}$$

$$y_j^- = \begin{cases} \min y_i & ; \text{ jika } j \text{ adalah atribut } b \\ \max y_i & ; \text{ jika } j \text{ adalah atribut } c_i \end{cases}$$

8. Menghitung jarak antara nilai setiap alternatif dengan matriks solusi ideal positif dan matriks solusi ideal negatif ( $D_i^+$  dan  $D_i^-$ ).

Dengan menggunakan persamaan :

$$D_i^- = \sqrt{\sum_{i=1}^n (x_i^+ - x_i^-)^2} \quad \text{persamaan (5)}$$

$$D_i^- = \sqrt{\sum_{j=1}^n (y_{ij} - y_i^-)^2} \quad \text{persamaan (6)}$$

9. Menghitung nilai preferensi untuk setiap alternatif ( $V_i$ ).

Nilai preferensi  $V_i$  untuk setiap alternatif dirumuskan :

$$V_i = \frac{D_i^-}{D_i^- + D_i^+} \quad \text{persamaan (7)}$$

## 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 lainnya disajikan dalam bentuk grafis (dalam format GIF, JPG, PNG, dll), suara (dalam format AU, WAV, dll), dan objek multimedia lainnya (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 Explorer* berbagai aplikasi *browser* lainnya.

## 2.7 MYSQL

(YM.Khusuma Ardhama 2012) MySQL dikembangkan oleh sebuah perusahaan Swedia bernama MySQL AB yang pada saat itu bernama TcX DataKonsult AB

sekitar tahun 1994-1995, namun cikal bakal kodenya sudah ada sejak 1979. Awalnya TcX membuat MySQL dengan tujuan mengembangkan aplikasi web untuk klien. TcX merupakan perusahaan pengembang *software* dan konsultan *database*. Saat ini MySQL sudah diakuisisi oleh Oracle Crop. MySQL adalah salah satu jenis *database server* yang sangat terkenal dan banyak digunakan untuk membangun aplikasi *web* yang mana *database* sebagai sumber dan pengelolaan datanya.

## 2.8 HTML

(Raharjo et al. 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>`.

## 2.9 PHP

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

Berbeda dengan HTML, kode PHP tidak diberikan secara 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 `?>`.

## 2.10 DFD (Data Flow Diagram)

(S & Shalahuddin 2016) *Data Flow Diagram* adalah representasi grafik yang menggambarkan aliran informasi dan transformasi 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 merepresentasikan 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).

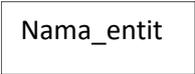
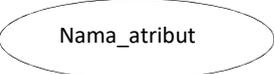
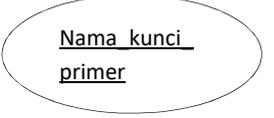
Notasi	Keterangan
	<p>Proses atau fungsi; pada pemodelan perangkat lunak pada pemrograman terstruktur, maka pemodelan notasi inilah yang harus menjadi prosedur didalam kode program.</p>
	<p><i>File</i> atau basis data atau penyimpanan (<i>storage</i>) pada pemodelan perangkat lunak,, maka pemodelan notasi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

## 2.11 ERD (Entity Relationship Diagram)

(S & Shalahuddin 2016) Pemodelan awal basis data yang paling banyak digunakan adalah *Entity Relationship Diagram* (ERD). ERD dikembangkan berdasarkan teori

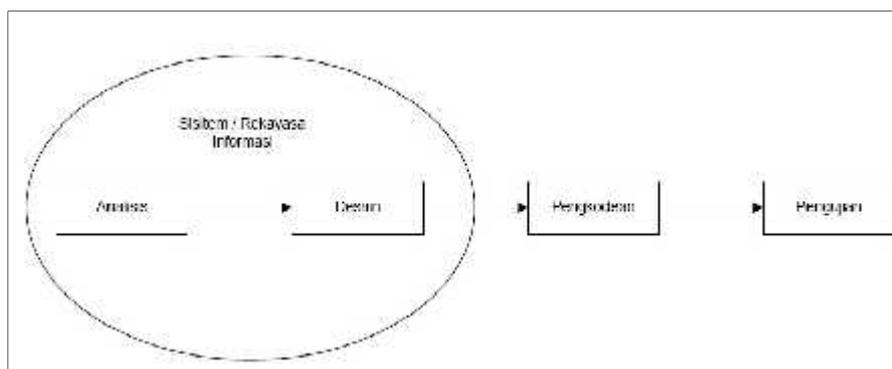
himpunan dalam bidang matematika. ERD digunakan untuk pemodelan basis data relasional. Notasi-notasi pada ERD adalah seperti pada Tabel 2.2.

Tabel 2.2 Notasi Pada Model ERD.

Simbol	Keterangan
Entitas /entity 	Entitas/entity merupakan data inti yang akan disimpan; bakal table pada basis data
Atribut 	Atribut/field; atau kolom data yang butuh disimpan dalam suatu entitas
Atribut Kunci Primer 	<i>Field</i> atau kolom digunakan sebagai kunci akses record yang diinginkan; biasanya berupa id
Atribut Multi nilai 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu.
Relasi 	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja.

## 2.12 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 & Shalahuddin 2016), yaitu:



Gamabr 2.7 Ilustrasi Model *Waterfall* Sumber (S & Shalahuddin 2016).

### 2.12.1 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*.

### 2.12.2 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.

### 2.12.3 Pengkodean

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

#### 2.12.4 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.

#### 2.12.5 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.

### 2.13 Penelitian Terdahulu

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

Tabel 2.3 Penelitian Terdahulu.

Nama	Judul	Terbit/ Tahun	Keterangan
Ikmah	Sistem Pendukung Keputusan Seleksi Penerimaan Dosen Menggunakan Metode TOPSIS	SNTIM, 2016; Yogyakarta.. ISSN : 2302-3805.	Metode <i>fuzzy</i> TOPSIS dapat digunakan dalam seleksi penerimaan dosen dengan menggunakan kriteria-kriteria yang telah ditetapkan menghasilkan rekomendasi calon dosen dengan ranking nilai tertinggi.

Tabel 2.3 (lanjutan).

S. Lestasi, W. Priyodiprodjo	Implementasi Metode <i>fuzzy</i> TOPSIS UNTUK Seleksi Penerimaan Karyawan	IJCCS, Vol.5 No.2, 2011; Bali.	Dengan adanya suatu sistem yang dapat memberikan rekomendasi sebagai bahan pertimbangan untuk pengambilan keputusan secara tepat yaitu dengan mengimplementasikan metode <i>fuzzy</i> TOPSIS dapat mengatasi permasalahan yang sering terjadi dalam perusahaan dalam menyaring pelamar pekerjaan
Hazwar Ambo Enre, Supeno Mardi S.N, Maudhi Hery Purnomo	Sistem Pendukung Keputusan Berbasis Data Multidimensi Menggunakan <i>Fuzzy</i> TOPSIS	Prosiding SMNT XXII, 2015; ITS, Surabaya.	Dengan penelitian Sistem Pendukung Keputusan Berbasis Data Multidimensi Menggunakan <i>Fuzzy</i> TOPSIS bertujuan untuk dapat menghasilkan suatu sistem pendukung keputusan sebagai rekomendasi terhadap evaluasi penawaran pada Sistem Pengadaan Barang dan Jasa Pemerintah Secara Elektronik (SPSE).

Tabel 2.3 (lanjutan).

Luthfi Nur Hidayat	Metode TOPSIS untuk Pemiliha Jurusan Pada Sekolah Menengah Atas	Universitas Dian Nusawantoro, Surabaya. 2015.	Pendukung keputusan menggunakan metode TOPSIS yang dapat membantu siswa SMA untuk pengambilan keputusan dalam pemilihan jurusan.
Awawin Mustana Rohmah	Seleksi Penerimaan Siswa Baru dengan Metode <i>Fuzzy</i> MADM TOPSIS	Universitas Brawijaya, Malang. 2015	Dengan menerapkan metode <i>fuzzy</i> TOPSIS, diperoleh hasil rekomendasi yang lebih akurat dan sesuai dengan ketentuan seleksi dalam pemilihan siswa baru yang berkualitas.