

## BAB II

### STUDI PUSTAKA

#### 2.1 *State Of The Art*

*State of the art* membahas perihal penelitian yang telah dilakukan dan diteliti oleh peneliti sebelumnya mengenai sistem pakar mendiagnosis gangguan kecemasan menggunakan metode *dempster shafer*. Adapun beberapa penelitian sebelumnya yang telah diteliti, diantaranya yakni :

Tabel 2. 1 *State of the art*

No	Peneliti	Judul	Metode	Pembahasan	Hasil Penelitian
1.	Nita Trisnowati, dkk (2014) [10]	Sistem Pakar Mendiagnosis Gangguan <i>Anxietas</i> Dengan Menggunakan <i>Teorema Bayes</i>	<i>Teorema Bayes</i>	Pada penelitian ini diterapkan metode <i>Teorema Bayes</i> yang dipakai untuk menghitung nilai probabilitas hasil identifikasi penyakit kecemasan.	Aplikasi ini layak digunakan untuk mendiagnosis gangguan <i>anxietas</i> . Sistem ini dapat digunakan oleh pakar psikolog sebagai pendamping dalam menangani gangguan <i>anxietas</i> pada pasiennya.
2.	Dewi Ayu Nur Wulandari (2015) [11]	Metode <i>Dempster Shafer</i> Pada Sistem Pakar Untuk Mendiagnosis Penyakit Perut	<i>Dempster Shafer</i>	metode <i>Dempster shafer</i> untuk menentukan jenis penyakit perut berdasarkan gejala yang di input. Teori <i>Dempster-Shafer</i> digunakan untuk mengkombinasikan bagian	Metode <i>Dempster-Shafer</i> berhasil diterapkan pada sistem pakar mendiagnosis penyakit perut yang digunakan untuk memprediksi jenis-jenis penyakit perut dengan memasukkan

Tabel 2. 2 *State of the art* (Lanjutan)

No	Peneliti	Judul	Metode	Pembahasan	Hasil Penelitian
				informasi yang terpisah untuk menghitung kemungkinan dari suatu kejadian	berupa gejala-gejala yang dimiliki pasien.
3.	Eva Yulia Puspaningrum, dkk (2016) [4]	Metode <i>Forward Chaining</i> Untuk Diagnosa Gangguan <i>Anxietas</i> Berbasis Mobile	<i>Forward Chaining</i>	Penelitian ini menggunakan metode <i>forward chaining</i> dan didukung dengan perhitungan probabilitas klasik. <i>Forward chaining</i> menggunakan teknik pencarian yang dimulai dengan menelusuri fakta yang ada dan mencocokkannya dengan aturan yang telah dibuat sehingga menghasilkan sebuah kesimpulan.	Berdasarkan uji validitas menggunakan tabel R didapatkan prosesentase sebesar 90% sesuai kebutuhan user. Dalam pembuatan aplikasi diagnosa kecemasan ini dapat membantu lebih dini dalam mendiagnosa gangguan kecemasan.
4.	Mikha Dayan Sinaga, dkk (2016) [12]	Penerapan Metode <i>Dempster Shafer</i> Untuk Mendiagnosis Penyakit Dari Akibat Bakteri <i>Salmonella</i>	<i>Dempster Shafer</i>	Pada penelitian ini Metode <i>Dempster Shafer</i> diterapkan untuk menghitung suatu nilai densitas suatu gejala terhadap penyakit yang disebabkan oleh bakteri <i>salmonella</i> .	Metode perhitungan <i>Dempster Shafer</i> mampu memberikan persentase nilai densitas dan rekomendasi perhitungan yang akurat untuk dapat dijadikan referensi diagnosis gangguan yang disebabkan bakteri <i>salmonella</i>

Tabel 2. 3 *State of the art* (Lanjutan)

No	Peneliti	Judul	Metode	Pembahasan	Hasil Penelitian
5.	Maura Widyaningih, dkk (2017) [13]	<i>Dempster Shafer</i> Untuk Sistem Diagnosis Gejala Penyakit Kulit Pada Kucing	<i>Dempster Shafer</i>	Penelitian ini mengenai penyakit kulit pada kucing, dengan penyelesaiannya menggunakan metode inferensi forward chaining dan perhitungan deteksi penyakit dengan dengan Dempster Shafer.	Sistem berhasil mengetahui jenis penyakit kulit kucing sesuai dengan yang diderita menerapkan teori <i>Dempster Shafer</i> untuk menghitung densitas dan memberikan solusi pengobatan sesuai penyakit
6.	Endang Lestari, dkk (2017) [14]	Sistem Pakar Dengan Metode <i>Dempster Shafer</i> untuk diagnosa gangguan Layanan Indihome (PT Telkom Magelang)	<i>Dempster Shafer</i>	Perhitungan <i>dempster shafer</i> dilakukan dari nilai belief dan nilai plausability yang diperoleh dari 1- nilai belief, dapat diketahui suatu gangguan dan nilai kepercayaan berdasarkan hasil perhitungan	Metode <i>dempster shafer</i> berhasil diimplementasikan dalam system yang digunakan untuk mendeteksi gangguan layanan Indihome.dengan menginput gejala yang dialami pelanggan melalui antarmuka sistem.
7.	Nita Sari Br Sembiring (2017) [15]	Penerapan Metode <i>Dempster Shafer</i> Untuk Mendiagnosis	<i>Dempster Shafer</i>	Implementasi <i>dempster shafer</i> dapat digunakan untuk melakukan dalam perhitungan kemungkinan untuk mendapat	Pada penelitian ini dapat ditarik kesimpulan bahwa semakin banyak rule yang diinputkan maka tingkat keakuratan hasil

Tabel 2. 4 *State of the art* (Lanjutan)

No	Peneliti	Judul	Metode	Pembahasan	Hasil Penelitian
		Penyakit Dari Akibat Bakteri Treponema Pallidum		Hasil diganosa yang sesuai dengan gejala yang diinput. Penelitian ini menggunakan metode inferensi dengan aturan telah disediakan, dimana semakin banyak aturan yang diinputkan maka tingkat keakurasian semakin kuat.	diagnosa semakin tepat.
8.	Raka Yusuf, dkk (2018) [16]	Aplikasi Diagnosa Gangguan Kecemasan Menggunakan Metode <i>Forward Chaining</i> Berbasis Web Dengan Php Dan Mysql	<i>Forward Chaining</i>	Aplikasi ini dimulai dengan menginputkan kumpulan fakta ke dalam sebuah memori kerja yang kemudian menurunkan fakta baru yang berlandaskan rule dimana premisnya cocok dengan fakta yang telah diketahui, proses dilanjutkan sampaimencapai solusi yang tepat dan sesuai.	Sistem telah berhasil dibangun sesuai dengan rencana dan perancangan yang sudah dilakukan, sistem ini dibuat untuk mempermudah dalam mendiagnosa dan memberikan solusi lebih dini dalam mengatasi gangguan kecemasan dengan cepat.
9.	Reza Setiawan, dkk (2018) [7]	Implementasi Metode <i>Dempster Shafer</i> Pada	<i>Dempster Shafer</i>	Pada penelitian ini menggunakan metode <i>dempster shafer</i>	Hasil pengujian sistem secara keseluruhan dengan Membandingkan

Tabel 2. 5 *State of the art* (Lanjutan)

No	Peneliti	Judul	Metode	Pembahasan	Hasil Penelitian
		Sistem Pakar Diagnosa Infeksi Penyakit Tropis Berbasis Web		Untuk menentukan hasil diagnosis dengan melihat nilai densitas yang muncul dari perhitungan. Nilai <i>belief</i> pada masing-masing gejala sangat berpengaruh terhadap hasil diagnosa penyakit. Semakin tinggi nilai <i>belief</i> pada sebuah gejala maka semakin besar kemungkinan seseorang menderita suatu penyakit berdasarkan gejala yang dialami tersebut.	Data hasil diagnosa sistem dan hasil diagnosa rekam medis rumah sakit RSUD. Sultan Syarif Mohammad Alkadrie, didapat tingkat akurasi keberhasilan sistem sebesar 94,23 %.
10.	Januar Dwie Amanda, dkk (2018) [17]	Implementasi Metode <i>Dempster-Shafer</i> untuk Mendeteksi Penyakit Diabetes Mellitus	<i>Dempster Shafer</i>	Aplikasi ini menerapkan metode <i>Dempster shafer</i> dimana gejala yang telah diberikan nilai densitas dari pakar digunakan untuk perhitungan dan penentuan hasil persentase sebuah gangguan.	Hasil pengujian keakuratan sistem untuk mendiagnosa penyakit <i>diabetes mellitus</i> menggunakan metode <i>dempster shafer</i> memiliki tingkat keakuratan presentase sebesar 81,81%.

## 2.2 Studi Literatur

### 2.2.1 Sistem Pakar

Sistem pakar merupakan komponen dari kecerdasan buatan, sistem pakar pertama kali dikembangkan oleh komunitas AI (*Artificial intelligence*) yaitu Edward Feigenbaum pada pertengahan tahun 1960 [18]. Sistem pakar berusaha mengambil sebuah pengetahuan dari manusia (pakar) yang direalisasikan ke dalam sebuah sistem komputer. Sistem pakar adalah suatu sistem perangkat lunak yang mengandung pengetahuan dirancang untuk menyelesaikan suatu masalah yang spesifik sebagaimana layaknya yang dikerjakan seperti seorang ahli [18]. Pakar adalah seseorang yang memiliki suatu keahlian atau pengetahuan yang spesifik sehingga dapat menyelesaikan suatu masalah yang biasanya tidak dapat dipecahkan bagi orang biasa [16].

Sistem pakar tersusun dalam 2 bagian pokok berupa lingkungan konsultasi dan pengembang. Lingkungan konsultasi digunakan untuk mendapatkan ilmu pengetahuan ahli yang dipakai oleh pengguna, sedangkan lingkungan pengembang digunakan untuk meinputkan sebuah ilmu pengetahuan pakar ke dalam sistem [11].

Adapun kelemahan dan juga kelebihan sistem pakar diantaranya berikut ini [18] :

#### 1. Kelebihan sistem pakar

- a. Membuat seseorang yang awam mengetahui lebih dini gangguan layaknya pakar.

- b. Meningkatkan produktivitas dampak meningkatnya kualitas hasil pekerjaan, karena dapat meningkatkan efisiensi kerja.
  - c. Menghemat waktu kerja, menyederhanakan serta memudahkan suatu pekerjaan.
  - d. Menyimpan keahlian dan pengetahuan para ahli.
  - e. Memiliki kemampuan untuk mengakses sebuah pengetahuan.
2. Kelemahan sistem pakar
- a. Sistem pakar tidak selalu 100% bernilai benar.
  - b. Sulit untuk dikembangkan karena terbatasnya ketersediaan ahli pakar pada bidangnya.

### 2.2.2 Metode Dempster Shafer

Metode *dempster shafer* merupakan sebuah metode dalam ilmu matematika dimana pembuktian yang berdasarkan nilai kepercayaan (*belief*) dan *plausible reasoning* (pemikiran yang masuk akal) yang digunakan untuk menggabungkan bagian-bagian informasi yang terpisah (bukti) untuk menghitung kemungkinan dari suatu kejadian. *Dempster shafer* adalah nilai parameter klinis yang dibagikan untuk memperlihatkan besarnya suatu nilai kepercayaan [7].

Teori ini dikembangkan pertama kali oleh dua ilmuwan yakni Arthur P. Dempster dan Glenn Shafer. Mereka melakukan sebuah penelitian berupa model ketidakpastian dengan *range* probabilitas sebagai probabilitas tunggal. Teori ini dipublikasikan pada buku yang memiliki judul *Mathematical Theory of Evident* diterbitkan pada tahun 1976 [19].

Secara luas metode *dempster shafer* dituliskan dalam sebuah interval yaitu [*Belief, Plausibility*]. Adapun penjelasannya akan diuraikan sebagai berikut [7] :

1. *Belief (Bel)*

*Belief* merupakan ukuran kekuatan gejala dalam mendukung sebuah gangguan atau himpunan proposisi. Dimana nilai suatu gejala diinput antara 0 sampai 1, jika gejala bernilai 0 maka menunjukkan tidak adanya gejala pada gangguan, dan jika bernilai 1 maka menunjukkan adanya suatu kepastian pada gejala. Fungsi *belief* dapat dinotasikan sebagai :

$$Bel(X) = \sum_{Y \subseteq X} m(Y)$$

2. *Plausibility (Pl)*, dirumuskan sebagai:

$$Pl(x) = 1 - Bel(x)$$

Dimana :

$$Bel(x) = Belief(x)$$

$$m(X) = \text{mass function dari (X)}$$

$$Pl(X) = Plausibility (X)$$

$$m(Y) = \text{mass function dari (Y)}$$

*Plausibility* bernilai dari 0 – 1 juga. Nilai *plausability* ini didapatkan dari 1 dikurangi dengan nilai *belief* suatu gejala. Jika yakin pada x, maka bisa dinyatakan bahwa  $Bel(x) = 1$ , dan  $Pl(x) = 0$  [7].



### 3. Environment

Teori *dempster shafer* dikenal dengan adanya *frame of discernment (FOD)* yang biasa disimbolkan dengan  $\Theta$  dan *mass function* yang disimbolkan dengan  $m$ . FOD adalah semesta pembicaraan dari kumpulan yang biasa juga disebut dengan *environment* [13] :

$$\Theta = \{\Theta_1, \Theta_2, \dots, \Theta_n\}$$

Dimana :

$\Theta$  = *environment* atau FOD dan

$\Theta_n$  = elemen bagian dalam *environment*

Environment berisi unsur-unsur yang mendeskripsikan kemungkinan sebagai jawaban dan hanya ada satu yang bisa selaras dengan jawaban yang akan diperlukan.

### 4. Aturan Kombinasi

Untuk menangani beberapa gejala pada *dempster shafer* menggunakan suatu ketentuan yang biasa disebut dengan *dempster's rule of combination*. Secara luas aturan kombinasi dirumuskan dengan berikut :

$$m_1 \oplus m_2(Z) = \frac{\sum_{x \cap y = z} m_1(x)m_2(y)}{1 - k}$$

Dimana:

$\oplus$  = operator *direct sum*

$m_1(x)$  = *mass function* dari *evidence* (X)

$m_2(y)$  = *mass function* dari *evidence* (Y)

$k$  = jumlah *evidential conflict*

Besarnya jumlah  $k$  (*evidential conflict*) dinotasikan sebagai :

$$k = \sum_{X \cap Y = \emptyset} m_1(X)m_2(Y)$$

Sehingga jika persamaan diatas disubstitusikan maka akan menjadi persamaan berikut :

$$m_3(Z) = \frac{\sum_{x \cap y = z} m_1(x).m_2(y)}{1 - \sum_{x \cap y = \emptyset} m_1(x).m_2(y)}$$

Keterangan :

$m_3(Z)$  = *mass function* dari *evidence* (Z)

### 2.2.3 Gangguan Kecemasan

Gangguan kecemasan merupakan suatu bagian gangguan psikologis dimana menyebabkan penderitanya mengalami rasa cemas dan panik yang besar dan berlebihan yang ditandai dengan rasa takut dan gejala fisik. Gangguan kecemasan juga dapat diartikan sebagai sekelompok kondisi yang memberikan gambaran tentang kecemasan yang berlebihan, disertai dengan emosional, perilaku, dan fisiologis [2]. Dua faktor yang dapat mempengaruhi gangguan kecemasan yakni internal dan eksternal. Faktor internal meliputi tipe kepribadian, *gender*, serta tingkat pendidikan. Sedangkan untuk faktor eksternal mencakup ancaman terhadap sistem diri, lingkungan sosial, maupun lingkungan keluarga [3].

Sebagian besar seseorang merasa cemas dalam menghadapi situasi yang mengancam atau tertekan. Gejala tersebut merupakan respons yang normal terhadap stress, akan tetapi kecemasan menjadi abnormal apabila kecemasannya berlebihan, sulit dikendalikan, tidak sesuai dengan tingkat keparaan stres, atau terjadi tanpa adanya stressor eksternal yang mengganggu kinerja hidup individu, baik itu pada kehidupan sosial ataupun keluarga. Gangguan kecemasan ini membuat seseorang memperlihatkan perilaku yang tidak lazim seperti menjadi tidak bahagia, pesimis, panik, takut yang terlepas dari ada atau tidak adanya bahaya [16]. Ada beberapa jenis gangguan kecemasan pada DSM 5 diantaranya yaitu sebagai berikut [20] :

1. *Generalized Anxiety Disorder*

*Generalized Anxiety Disorder* atau dapat diterjemahkan dengan Gangguan kecemasan umum merupakan kekhawatiran yang menetap (bertahan lama) yang sering kali terjadi terhadap hal-hal kecil dan disertai dengan gejala somatik yang menyebabkan gangguan fungsi sosial dan fungsi pekerjaan atau perasaan, nyeri hebat, serta perasaan tidak enak [1]. Gangguan ini disertai setidaknya dengan 3 gejala psikologis dan fisik seperti mudah lelah, konsentrasi menurun, insomnia, otot tegang, mudah marah dan gelisah, Sulit mengendalikan perasaan cemas, dan kecemasan ini biasanya bukan berasal dari pengaruh obat [21].

2. Gangguan Panik

Gangguan Panik merupakan gangguan kecemasan yang ditandai dengan serangan panik spontan yang biasanya terjadi dalam hitungan menit (10-20

menit), yang mencakup simtom-simtom fisiologis (detak jantung cepat, sesak napas, panas dan berkeringat, nyeri dada, kram perut, gemetar, pusing, merasa ingin pingsan) secara mendadak serta berulang dalam waktu sekurangnya selama 1 bulan [1]. Gejala lainnya adalah ketakutan akan kehilangan kontrol, muncul perasaan bahwa dirinya dan lingkungan sekitar tidak nyata (derealisasi), gangguan ini juga tidak berasal dari pengaruh obat [20].

### 3. Gangguan Fobia Spesifik

Gangguan fobia spesifik adalah ketakutan yang menetap hebat dan penolakan yang menetap dan tidak masuk akal terhadap sebuah objek, situasi ataupun aktivitas yang menimbulkan suatu keinginan untuk menjauh dan menghindari sesuatu yang ditakutinya. Individu menyadari bahwa rasa takutnya merupakan sesuatu yang berlebih atau yang tidak masuk akal terhadap bahaya yang aktual dari sebuah objek, aktivitas atau situasi [22]. Gejala lainnya adalah selalu menghindari objek/situasi yang ditakutinya, objek/situasi tersebut menyebabkan kecemasan yang tiba-tiba dan gejala ini diderita sekurangnya dalam waktu 6 bulan [20].

### 4. Gangguan Fobia Sosial

Gangguan fobia sosial adalah ketakutan yang irasional dan menetap dalam jangka waktu yang lama biasanya berhubungan dengan kehadiran orang lain. Individu yang mengalami ini lebih memilih menghindari dari situasi dimana mungkin akan dikritik yang membuat ia merasa dipermalukan atau terhina, dan menunjukkan tanda-tanda kecemasan atau menampilkan

perilaku yang memalukan sehingga dipandang negatif tapi pada kenyataannya tidak [22]. Gejala lain adalah saat berhadapan dengan situasi sosial muncul kecemasan tiba-tiba dan gejala juga penghindaran tersebut sangat mengganggu aktivitas keseharian individu [20].

#### 5. Gangguan Agorafobia

Gangguan adalah gangguan kecemasan dimana penderita akan menghindari dan menjauh dari berbagai situasi keramaian yang mungkin menyebabkan dirinya panik dan tidak bisa melarikan diri seperti dalam keramaian. Biasanya gejalanya adalah penderita tidak mau keluar rumah kecuali ada orang yang menemaninya untuk pergi keluar [20].

#### 2.2.4 Website

*Website* adalah sebuah aplikasi dengan informasi yang berisi dokumen multimedia yang meliputi teks, suara, gambar, video, animasi dan lainnya dalam sebuah *server* web internet yang memakai protokol HTTP (*Hypertext Transfer Protocol*), *website* diakses memakai perangkat lunak *browser*. *Browser* merupakan aplikasi yang sanggup mengelola dokumen web dengan cara diterjemahkan [23]. Proses browser dikerjakan pada bagian yang terdapat pada suatu aplikasi *browser* yang biasa disebut *web engine*. Situs web adalah dokumen-dokumen yang terkumpul menjadi satu kesatuan yang mempunyai URL (*Unified Resource Locator*) atau domain lalu dipublikasikan pada internet [24]. Dilihat dari segi konten, web tersusun menjadi dua macam yakni web statis dan dinamis. Adapun penjabaran mengenai jenis web adalah sebagai berikut ini [25] :

### 1. Web Statis

Web statis merupakan halaman web yang tampilan atau kontennya tidak bisa atau sulit dirubah. Jika ingin mengubah tampilannya maka harus dilakukan secara manual, yaitu dengan cara mengedit secara langsung file mentah struktur webnya atau kode-kodenya.

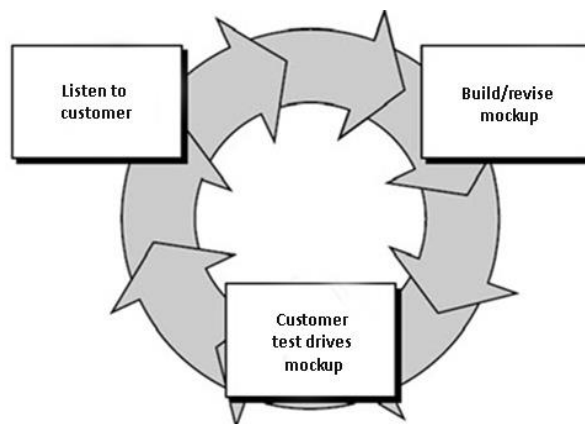
### 2. Web Dinamis

Web dinamis adalah web dengan tampilan atau kontennya bisa dirubah setiap saat, yang digunakan untuk menampilkan *update* tampilan sesering mungkin. Website dinamis bisa disesuaikan dengan kebutuhan, baik dari segi fitur ataupun dari segi tampilannya.

#### 2.2.5 Metodologi *Prototype*

*Prototype* merupakan metodologi pengembangan perangkat lunak yang perencanaannya dilakukan dengan cepat dimana terdapat umpan balik yang berulang, perbaikan sistem yang sesuai kebutuhan *user-interface* dan mementingkan pendekatan aspek desain dan fungsi. Metode ini memberikan fasilitas bagi pengembang dan pengguna untuk saling berkomunikasi selama tahap pembuatan sistem, sehingga pengembang bisa dengan mudah memodelkan sistem yang akan dibuat. Keunggulan *prototype* diantaranya adalah pengembang dan pelanggan dapat menjalin komunikasi yang baik, pelanggan dapat berperan aktif dalam proses pengembangan aplikasi. Sedangkan kekurangannya tidak cocok untuk diaplikasikan pada sebuah sistem yang sangat besar, mengakhirkan alternatif pemecahan masalah dan tidak

selamanya metode ini mudah dirubah [9]. Berikut adalah gambar tahapan dari *prototype* :



Gambar 2. 1 Tahapan *Prototype* [26]

Tahapan-tahapan yang terdapat pada metode *prototype* yang dapat dilihat pada Gambar 2.1 akan dijelaskan berikut [26] :

1. *Listen to customer*

Pada tahapan ini dilakukan wawancara dengan seorang narasumber (pakar) untuk mengumpulkan materi yang dibutuhkan dalam membangun perangkat lunak bantu untuk mendeteksi gangguan kecemasan. Setelah materi terkumpul maka akan dilanjutkan perancangan sistem.

2. *Build / revise mockup*

Pada tahapan ini dilakukan perancangan basis data, desain tampilan dan juga melakukan pengkodean dan yang akan diimplementasikan pada perangkat lunak. Kemudian akan dilakukan evaluasi pada aplikasi yang sedang dibangun untuk dapat mengurangi resiko kesalahan pada aplikasi.

3. *Customer test drives mockup*

Pada tahapan ini dilakukan pengujian terhadap perangkat lunak yang

sudah dibangun, pengujian aplikasi ini akan dilakukan oleh pengembang dan juga narasumber guna mengetahui keakurasian aplikasi.

### 2.2.6 DFD

DFD singkatan dari *Data Flow Diagram* adalah alat perancangan yang menggunakan simbol-simbol untuk menggambarkan atau menjelaskan suatu proses pada sistem atau perangkat lunak [27]. DFD memperlihatkan gambaran tentang input, proses dan output dari suatu sistem yaitu objek data yang mengalir pada sebuah sistem, lalu ditransformasi pada komponen pemrosesan dan objek data yang hasilnya akan keluar dari sistem [28].

DFD dapat mempermudah pemakainya yang sedikit menguasai sistem untuk memahami sistem yang akan dikerjakan. DFD biasanya digambarkan dalam bentuk struktur (hierarki) yang direpresentasikan dengan menggunakan tanda panah yang berlabel dan transformasi yang disimbolkan dengan lingkaran-lingkaran (*bubble* proses), DFD yang pertama biasanya disebut dengan DFD level 0 biasa disebut diagram konteks yaitu yang menggambarkan sistem secara keseluruhan lalu DFD berikutnya urutan dari DFD sebelumnya, adapun urutan dalam DFD akan dijelaskan berikut [28]:

1. Diagram Konteks (*Context Diagram*)

Diagram konteks biasanya disebut dengan diagram *zero* (diagram level 0) yakni diagram yang menggambarkan secara umum aliran input dan output informasi dan data yang akan dilakukan pada sistem atau perangkat lunak yang akan dirancang. Diagram ini menampilkan sistem yang akan

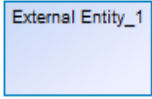

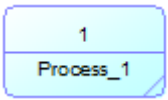
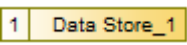


dirancang secara global, seluruh entitas eksternal harus digambarkan, sehingga dapat terlihat data yang mengalir pada input, proses dan output. Diagram konteks hanya terdiri dari satu proses saja dan proses saja tidak boleh lebih.

## 2. Diagram Rinci (*Level Diagram*)

Diagram rinci adalah diagram yang mendeskripsikan atau menggambarkan lebih spesifik lagi dari pada proses yang terdapat dalam diagram *zero*. *Data flow* yang keluar dan yang masuk dari sistem harus sama dengan yang ada pada diagram *zero*. Level 1 memaparkan proses-proses utama yang terdapat pada sistem yang dibangun, lalu DFD level 2 adalah pemaparan lebih rinci dari DFD level 1, setiap proses pada DFD level 1 bisa dimodelkan secara lebih spesifik menjadi sebuah DFD lainnya. Jika diperlukan, setiap proses yang ada dalam level 2 juga bisa diperinci menjadi level 3, dan seterusnya. Perincian DFD akan berhenti sampai proses sudah terperinci dan tidak dapat diperinci lagi.

Tabel 2. 6 Simbol-simbol pada DFD

Simbol	Komponen	Kegunaan
	External entity	Untuk menunjukkan subjek dilingkungan luar sistem yang memberi atau menerima input
	Data flow	Untuk menunjukkan arus dari data
	Proses	Untuk menunjukkan kegiatan atau kerja yang dilakukan oleh sistem
	Data store	Untuk menunjukkan simpanan dari data

### 2.2.7 *Process Specification (P-Spec)*

*Process Specification (P-spec)* adalah suatu penjelasan setiap proses yang terdapat pada DFD atau *data flow diagram*. P-spec ini digunakan untuk mendeskripsikan semua proses model aliran yang terlihat sampai pada tingkat akhir penyaringan. Isi dari P-spec terdiri dari teks naratif, nomor, nama proses, entitas, *data store*, *input*, *output* dan algoritma proses [29].

### 2.2.8 **Kamus Data**

Kamus data merupakan kumpulan simbol atau elemen yang mengalir pada sebuah sistem untuk membantu dalam mengidentifikasi ataupun menggambarkan setiap field yang terdapat pada basis data dalam sistem. Kamus data sangat menunjang analisis sistem dalam mendeskripsikan data yang mengalir dalam sistem, sehingga penjabaran dan penjelasan mengenai data bisa dilakukan secara terstruktur dan lengkap.

Kamus data dipakai sebagai alat komunikasi antara pengguna dan analisis sistem, pada analisis sistem yang berkenaan dengan data yang mengalir pada sistem dan pada perancangan sistem, pembuatan kamus data ini biasanya diambil pada *data store* yang terdapat di dalam *Data Flow Diagram (DFD)* yang akan dirancang. Kamus data biasanya menggunakan notasi-notasi atau simbol-simbol agar mempermudah untuk membacanya, adapun notasi-notasi pada kamus data dijelaskan berikut [8].

Tabel 2. 7 Notasi pada Kamus Data

Notasi	Fungsi
=	Disusun dari, terdiri dari, terbentuk dari
+	Dan
	Atau
[ ]	Pilih salah satu dari beberapa alternatif (pilihan)
{ }n	Pengulangan sebanyak n kali
( )	Data <i>optional</i>
@	Key field
**	Pembatas komentar

### 2.2.9 ERD

ERD singkatan dari *Entity Relationship Diagram* adalah suatu rancangan diagram pada sebuah basis data untuk menghasilkan diagram konseptual untuk model data sematik sistem [30]. ERD juga dapat menggambarkan objek-objek atau data yang dibentuk berdasarkan persepsi di dalam dunia nyata yang disebut dengan entitas (*entity*) dan hubungan antar entitas dengan entitas lainnya (relasi) yang menggunakan sejumlah notasi. Entitas harus bersifat unik dan memiliki atribut yang digunakan sebagai pembeda antara entitas lain [31]. ERD gambarkan dalam dua model yakni conceptual dan physical data model, berikut adalah penjelasannya.

#### 1. *Conceptual Data Model*

*Conceptual data model* yaitu model yang merepresentasikan tabel yang merupakan entitas yang didalamnya berisi atribut yang terdapat hubungan (relasi) antara entitas. Manfaat CDM adalah memberi gambaran yang lengkap dan lebih detail dari struktur bentuk *database* [32].

## 2. *Physical Data Model*

*Physical data model* (PDM) adalah model yang merepresentasikan beberapa tabel yang terstruktur untuk menggambarkan data dan hubungan antara data dengan data lain. Setiap tabel memiliki kolom dimana setiap kolomnya memiliki nama yang unik atau *primary key* [32].

### 2.2.10 PHP

PHP atau *Hypertext Preprocessor* adalah bahasa pemrograman yang khusus untuk pembangunan dan pengembangan aplikasi yang berbasis web yang berupa script yang ditempatkan pada server dan diproses dalam server, kemudian hasil dikirimkan kepada klien. Fungsi PHP diantaranya adalah meringkas script bahasa pemrograman, memasukkan data ke dalam *database*, melakukan manajemen *cookie* dan *session*. PHP bisa diintegrasikan dengan html, javascript, JQuery, dan ajax. Akan tetapi kebanyakan yang menggunakan PHP digunakan bersamaan dengan dokumen yang bertipe HTML [23].

PHP dibentuk sebagai *server-side scripting* yang bersatu dengan HTML untuk membangun halaman web dinamis yakni halaman web yang tampilan sesuai dengan perintah baik dari segi tampilan atau fitur. Dikarenakan PHP di desain sebagai *server-side scripting* maka script PHP dieksekusi pada server kemudian hasil yang dibentuk dalam format html dikirimkan ke dalam *browser*. Salah satu keunggulan PHP adalah bisa menciptakan suatu halaman yang dinamis karena keahliannya dalam melakukan koneksi di berbagai jenis perangkat lunak DBMS (*Database Management System*), PHP bisa digunakan pada berbagai jenis sistem operasi serta bisa dijalankan secara *runtime* melalui

*console*, dan PHP ini bisa digunakan dengan free atau dengan kata lain tidak memerlukan biaya untuk menggunakan perangkat lunak ini [25].

### 2.2.11 CSS

*Cassading Style Sheet* atau CSS adalah bahasa style sheet yang ditulis dalam bahasa markup untuk mengatur tampilan pada suatu dokumen dan digunakan untuk melekatkan *style* (minimum *font* atau pengaturan halaman) ke dalam dokumen yang tersusun contohnya dokumen HTML dan XML. Penggunaan CSS yang paling biasa digunakan adalah memformat halaman web yang ditulis dengan HTML. Dalam penyusunan file CSS terdapat tiga dasar yakni elemen, *class* dan id. Elemen berfungsi untuk menentukan semua elemen HTML dalam situs web. Id berfungsi untuk menentukan bagian yang hanya ada satu pada sebuah halaman dan menentukan *stylenya* dan hanya dapat dipanggil satu kali. Sedangkan *class* berfungsi untuk menentukan *style* yang bisa dipanggil beberapa kali pada satu halaman [33].

### 2.2.12 Java Script

*Javascript* yaitu bahasa pemrograman berupa sekumpulan *script* yang fungsinya berjalan pada dokumen HTML yang ditampilkan pada *browser* agar menjadi lebih interaktif. *Javascript* tidak memerlukan *compiler* untuk menjalankan programnya hanya cukup menggunakan *interpreter*. *Internet explorer* merupakan salah satu *browser* yang dilengkapi dengan interpreter *Java Script*. Aplikasi text editor yang dipakai untuk megimplementasikan bahasa *java script* dan web *browser* yang dipakai pada javascript harus

mempunyai fitur client-side, *high level programming language*, dan berorientasi objek.

*Javascript* membagikan kemampuan tambahan pada HTML dengan memberikan eksekusi perintah pada bagian *user* yang artinya pada bagian *browser* bukan pada bagian *server web*. *Javascript* dibuat untuk membentuk web yang interaktif. Setiap orang bisa memakai *Javascript* tanpa harus membayar lisensi [34].

### 2.2.13 Bootstrap

Bootstrap adalah aplikasi yang siap pakai untuk membuat sebuah *front-end* pada sebuah web yang mengutamakan tampilan untuk mobile device agar mempermudah dan mempercepat pengembangan web, dengan bootstrap dapat membangun web yang statis ataupun dinamis. Bootstrap menyediakan HTML, CSS dan *javascript* yang siap untuk digunakan. Bisa diartikan bootstrap adalah *template* desain web dengan fitur plus [35].

Bootstrap ini diciptakan guna mempermudah proses dalam desain sebuah web. Bootstrap merupakan sebuah *framework* untuk membentuk tampilan web secara reponsif yaitu tampilan web yang akan menyesuaikan ukuran layar dari browser yang digunakan baik dalam bentuk desktop ataupun mobile. Keunggulan bootstrap salah satunya menghemat waktu pembuatan font-end web dan bootstrap juga dibangun secara terstruktur sehingga web dapat diakses menjadi lebih ringan dari sebelumnya [35].

#### 2.2.14 MySQL

MySQL atau *My Structure Query Language* adalah aplikasi dari DBMS, yakni sebuah server basis data yang memakai bahasa sql sebagai dasar dalam mengolah atau menggunakan basis data dan juga banyak digunakan untuk membangun sebuah perangkat lunak yang berbasis web. MySQL merupakan aplikasi basisdata yang pertama kali dibantu memakai bahasa pemrograman script untuk internet dengan memakai *script* php dan perl untuk tersambung dengan internet. MySQL bersifat open source dan bisa digunakan secara free tanpa harus membayar lisensi untuk menggunakan MySQL [34].

Keunggulan yang dimiliki MySQL diantaranya yaitu probabilitas dimana basis data ini bisa berjalan dengan stabil pada beberapa platform sistem operasi yang biasa digunakan seperti Windows, OS/2, linux, dan berbagai versi unix lain, mempunyai lisensi GPL dan dapat diintegrasikan menggunakan beberapa bahasa pemrograman seperti java, net, perl, dan python. Salah satu kekurangannya adalah kurang mendukung pada bahasa pemrograman visual basic, delphi, dan lainnya [25].

#### 2.2.15 Software Testing

Pengujian perangkat lunak atau *Software Testing* merupakan proses pengecekan atau pemeriksaan pada suatu sistem dengan makna menemukan atau menyertakan kesalahan setiap kegiatannya yang bertujuan untuk mengevaluasi atribut kemampuan suatu sistem dan menentukan apakah sistem itu memenuhi hasil yang dibutuhkan atau tidak. Pengujian sangat diutamakan untuk memastikan sistem yang telah dibuat agar dapat berjalan sesuai dengan

fungsionalitas dan sesuai yang diharapkan yang berfungsi juga untuk menemukan ketidaksesuaian kesalahan pada sistem yang telah dibangun dengan perancangan sebelum sistem diserahkan kepada pengguna. Ada beberapa pendekatan pengujian perangkat lunak diantaranya adalah sebagai berikut [36] :

1. Pengujian *Black-Box*

Pengujian *blackbox* mengacu pada pengujian fungsionalitas perangkat lunak, metode pengujian menekankan pada evaluasi tampilan luar sistem, pemeriksaan fungsi dan pemeriksaan input dan Data Keluar tanpa harus mengetahui yang terjadi dalam proses atau tanpa harus menguji kode program secara rincinya. Pengujian *blackbox* berupaya menemukan kesalahan-kesalahan seperti fungsi-fungsi pada sistem yang salah, kesalahan antarmuka, kinerja, kesalahan pada struktur data dan lainnya untuk dijadikan evaluasi ke depan [36].

2. Pengujian *Black-Box*

Persamaan yang digunakan untuk mengetahui hasil dari tingkat akurasi sistem pada kasus pengujian.

$$\text{Nilai Keakuratan} = \frac{\text{Jumlah yang sesuai}}{\text{jumlah kasus}} \times 100\%$$