

BAB II

KAJIAN PUSTAKA

2.1 Teori Dasar

2.1.1 Kecerdasan Buatan

Menurut Sutojo, dkk. (2011: 1) kecerdasan buatan yang dimaksud itu merujuk pada mesin yang mampu berpikir, menimbang tindakan yang akan diambil, dan mampu mengambil keputusan seperti yang dilakukan oleh manusia. Persoalan-persoalan yang ditangani oleh kecerdasan buatan makin lama makin berkembang sehingga memungkinkan bagi kecerdasan buatan untuk merambah ke bidang ilmu yang lain. Hal ini disebabkan karakteristik cerdas sudah mulai dibutuhkan di berbagai disiplin ilmu dan teknologi.

Beberapa bidang yang menggunakan kecerdasan buatan antara lain sebagai berikut.

1. Logika Fuzzy (*Fuzzy Logic*)

Konsep tentang logika fuzzy diperkenalkan oleh Prof. Lotfi Astor Zadeh pada 1962. Logika fuzzy adalah metodologi sistem kontrol pemecahan masalah, yang cocok untuk diimplementasikan pada sistem, mulai dari sistem yang sederhana, sistem kecil, embedded system, jaringan PC, multi-channel atau workstation berbasis akuisisi data dan

sistem kontrol. Metodologi ini dapat diterapkan pada perangkat keras, perangkat lunak, atau kombinasi keduanya (Sutojo, dkk., 2011: 211).

Menurut Sutojo, dkk. (2011: 233) terdapat beberapa metode dalam logika fuzzy antara lain:

a. Metode Tsukamoto

Secara umum bentuk model fuzzy Tsukamoto adalah:

If (X IS A) and (Y IS B) Then (Z IS C)

Di mana A, B dan C adalah himpunan fuzzy.

b. Metode Mamdani

Metode Mamdani paling sering digunakan dalam aplikasi-aplikasi karena strukturnya yang sederhana, yaitu menggunakan operasi MIN-MAX atau MAX-PRODUCT.

c. Metode Sugeno

Dalam metode Sugeno, output sistem berupa konstanta atau persamaan linear. Secara umum bentuk model fuzzy Sugeno adalah:

IF $(x_1 \text{ is } A_1) \text{ and } \dots \text{ and } (x_N \text{ is } A_N)$ THEN $z = f(x,y)$

2. Jaringan Saraf Tiruan (JST)

Menurut Sutojo, dkk. (2011: 283), jaringan saraf tiruan adalah paradigma pengolahan informasi yang terinspirasi oleh sistem saraf secara biologis, seperti proses informasi pada otak manusia. Elemen kunci dari paradigma ini adalah struktur dari sistem pengolahan informasi yang terdiri dari sejumlah besar elemen pemrosesan yang

saling berhubungan (neuron), bekerja serentak untuk menyelesaikan masalah tertentu.

3. Sistem Pakar (*Expert System*)

Menurut Sutojo, dkk. (2011: 159) sistem pakar merupakan cabang dari *Artificial Intelligent* (AI) yang cukup tua karena sistem ini mulai dikembangkan pada pertengahan 1960. Istilah sistem pakar berasal dari istilah *knowledge-based expert system*. Istilah ini muncul karena untuk memecahkan masalah, sistem pakar menggunakan pengetahuan seorang pakar yang dimasukkan ke dalam computer. Beberapa metode dalam sistem pakar adalah sebagai berikut.

a. *Forward Chaining*

Forward Chaining adalah teknik pencarian yang dimulai dengan fakta yang diketahui, kemudian mencocokkan fakta-fakta tersebut dengan bagian *IF* dari *rules IF-THEN*.

b. *Backward Chaining*

Backward Chaining adalah metode inferensi yang bekerja mundur ke arah kondisi awal. Proses diawali dari *Goal* (yang berada dibagian *THEN* dari *rule IF-THEN*), kemudian pencarian mulai dijalankan untuk mencocokkan apakah fakta-fakta yang ada cocok dengan premis-premis di bagian *IF*.

2.1.2 Sistem Pakar

Menurut Hayadi (2016: 1) sistem pakar atau *Expert System* biasa disebut juga dengan *Knowledge Based System* yaitu suatu aplikasi komputer yang ditujukan untuk membantu pengambilan keputusan atau pemecahan persoalan dalam bidang yang spesifik. Sistem ini bekerja dengan menggunakan pengetahuan dan metode analisis yang telah didefinisikan terlebih dahulu oleh pakar yang sesuai dengan bidang keahliannya. Sistem ini disebut dengan sistem pakar karena fungsi dan perannya sama seperti seorang ahli yang harus memiliki pengetahuan, pengalaman dalam memecahkan suatu persoalan. Sistem biasanya berfungsi sebagai kunci penting yang akan membantu suatu sistem pendukung keputusan atau sistem pendukung eksekutif.

Menurut Sutojo, dkk. (2011: 163) pakar adalah seorang yang mempunyai pengetahuan, pengalaman, dan metode khusus, serta mampu menerapkannya untuk memecahkan masalah atau memberi nasihat. Seorang pakar harus mampu menjelaskan dan mempelajari hal-hal baru yang berkaitan dengan topik permasalahan, jika perlu harus mampu menyusun kembali pengetahuan-pengetahuan yang didapatkan, dan dapat memecahkan aturan-aturan serta menentukan relevansi kepakarannya. Jadi seorang pakar harus mampu melakukan kegiatan-kegiatan berikut:

1. Mengenali dan memformulasikan permasalahan.
2. Memecahkan permasalahan secara cepat dan tepat.
3. Menerangkan pemecahannya.
4. Belajar dari pengalaman.

5. Merestrukturisasi pengetahuan.
6. Memecahkan aturan-aturan.
7. Menentukan relevansi.

Menurut Sutojo, dkk. (2011: 160) sistem pakar menjadi sangat populer karena sangat banyaknya kemampuan dan manfaat yang diberikan, di antaranya:

1. Meningkatkan produktivitas, karena sistem pakar dapat bekerja lebih cepat daripada manusia.
2. Membuat seorang yang awam bekerja seperti layaknya seorang pakar.
3. Meningkatkan kualitas, dengan memberi nasehat yang konsisten dan mengurangi kesalahan.
4. Mampu menangkap pengetahuan dan kepakaran seseorang.
5. Dapat beroperasi di lingkungan yang berbahaya.
6. Memudahkan proses pengetahuan seorang pakar.
7. Andal. Sistem pakar tidak pernah menjadi bosan dan kelelahan atau sakit.
8. Meningkatkan kapabilitas sistem komputer. Integrasi sistem pakar komputer lain membuat sistem lebih efektif dan mencakup lebih banyak aplikasi.
9. Mampu bekerja dengan informasi yang tidak lengkap atau tidak pasti. Berbeda dengan sistem komputer konvensional. Sistem pakar dapat bekerja dengan informasi yang tidak lengkap. Pengguna dapat merespons dengan: “tidak tahu” atau “tidak yakin” pada satu atau lebih pertanyaan selama konsultasi dan sistem pakar tetap akan memberikan jawabannya.

10. Bisa digunakan sebagai media pelengkap dan pelatihan. Pengguna pemula yang bekerja dengan sistem pakar akan menjadi lebih berpengalaman karena adanya fasilitas penjelas yang berfungsi sebagai guru.
11. Meningkatkan kemampuan untuk menyelesaikan masalah karena sistem pakar mengambil sumber pengetahuan dari banyak pakar.

Selain manfaat, ada juga beberapa kekurangan yang ada pada sistem pakar, diantaranya:

1. Biaya yang sangat mahal untuk membuat dan memeliharanya.
2. Sulit dikembangkan karena keterbatasan keahlian dan ketersediaan pakar.
3. Sistem pakar tidak 100% bernilai benar.

Menurut Hayadi (2016: 3) ciri-ciri dari sistem pakar adalah sebagai berikut:

1. Terbatas pada *domain* keahlian tertentu.
2. Dapat memberikan penalaran untuk data yang tidak pasti.
3. Dapat mengemukakan rangkaian alasan yang diberikannya dengan cara yang dapat dipahami.
4. Berdasarkan pada kaidah atau *rule* tertentu.
5. Dirancang untuk dapat dikembangkan secara bertahap
6. Pengetahuan dan mekanisme inferensi jelas terpisah.
7. Keluarannya bersifat anjuran.
8. Sistem dapat mengaktifkan kaidah secara searah yang sesuai yang dituntun oleh dialog dengan pemakai.

2.1.2.1 Komponen Sistem Pakar

Menurut Hayadi (2016: 6) sebuah program sistem pakar terdiri atas beberapa komponen yang mutlak harus ada. Komponen itu adalah sebagai berikut:

1. Basis Pengetahuan (*Knowledge Base*)

Basis pengetahuan merupakan inti program sistem pakar karena basis pengetahuan ini merupakan representasi pengetahuan (*Knowledge Representation*) dari seorang pakar. Menurut Hartati dan Iswanti (2008: 5), basis pengetahuan merupakan kumpulan pengetahuan bidang tertentu pada tingkatan pakar dalam format tertentu. Pengetahuan ini diperoleh dari akumulasi pengetahuan pakar dan sumber-sumber pengetahuan lainnya seperti yang telah disebutkan sebelumnya. Basis pengetahuan bersifat dinamis, bisa berkembang dari waktu ke waktu. Pada sistem pakar basis pengetahuan terpisah dari mesin inferensi. Pemisahan ini bermanfaat untuk pengembangan sistem pakar secara leluasa disesuaikan dengan perkembangan pengetahuan pada suatu domain.

2. Basis Data

Basis data adalah bagian yang mengandung semua fakta, baik fakta awal pada saat sistem mulai beroperasi maupun fakta yang didapatkan pada saat pengambilan kesimpulan sedang dilaksanakan.

3. Mesin Inferensi

Mesin inferensi adalah bagian yang mengandung mekanisme fungsi berfikir dan pola penalaran sistem yang digunakan oleh seorang pakar. Mekanisme ini akan menganalisa suatu masalah tertentu dan selanjutnya akan mencari jawaban atau kesimpulan yang terbaik. Mesin inferensi memulai pelacakannya dengan mencocokkan kaidah dalam basis pengetahuan dengan fakta yang ada dalam basis data. Ada dua teknik inferensi yang ada yaitu pelacakan ke belakang (*Backward Chaining*) yang memulai penalaran dari kesimpulan hipotesa menuju fakta yang mengandung hipotesa tersebut. Dan yang kedua yakni pelacakan ke depan (*Forward Chaining*) yang merupakan kebalikan dari pelacakan ke belakang yaitu memulai dari sekumpulan data menuju kesimpulan. Menurut Hartati dan Iswanti (2008: 5), mesin inferensi sesungguhnya adalah program computer yang menyediakan metodologi untuk melakukan penalaran tentang informasi pada basis pengetahuan dan pada memori kerja, serta untuk merumuskan kesimpulan-kesimpulan. Komponen ini menyajikan arahan-arahan tentang bagaimana menggunakan pengetahuan dari sistem dengan membangun agenda yang mengelola dan mengontrol langkah-langkah yang diambil untuk menyelesaikan masalah ketika dilakukan konsultasi.

4. Antar Muka Pemakai (*User Interface*)

Antar muka pemakai adalah bagian penghubung antara program sistem pakar dengan pemakainya. Pada bagian ini akan terjadi dialog antara

program dengan pemakai. Program akan mengajukan pertanyaan berbentuk “ya/tidak” (yes or no question) atau berbentuk menu pilihan. Melalui jawaban yang diberikan oleh pemakai, sistem pakar akan mengambil keputusan yang berupa informasi ataupun anjuran sesuai dengan sifat dari sistem pakar. Menurut Hartati dan Iswanti (2008: 4), antar muka yang efektif dan ramah pengguna (*user-friendly*) penting sekali terutama bagi pemakai yang tidak ahli dalam bidang yang diterapkan pada sistem pakar.

2.1.3 *Certainty Factor* (Faktor Kepastian)

Teori *Certainty Factor* (CF) diusulkan oleh Shortliffe dan Buchanan pada 1975 untuk mengakomodasi ketidakpastian pemikiran (inexact reasoning) seorang pakar. Seorang pakar, (misalnya dokter) sering kali menganalisis informasi yang ada dengan ungkapan seperti “mungkin”, “kemungkinan besar”, “hampir pasti”. Untuk mengakomodasi hal ini kita menggunakan *certainty factor* (CF) guna menggambarkan tingkat keyakinan pakar terhadap masalah yang sedang dihadapi (Sutojo, dkk., 2011: 194).

Terdapat dua cara dalam mendapatkan tingkat keyakinan (CF) dari sebuah rule, yaitu:

1. Metode ‘*Net Belief*’ yang diusulkan oleh E.H. Shortliffe dan B. G. Buchanan

$$CF(Rule) = MB(H,E) - MD(H,E)$$

$$MB(H,E) = \begin{cases} 1 & P(H) = 1 \\ \frac{\max[P(H|E), P(H)] - P(H)}{\max[1,0] - P(H)} & \text{lainnya} \end{cases}$$

$$MD(H,E) = \begin{cases} 1 & P(H) = 0 \\ \frac{\min[P(H|E), P(H)] - P(H)}{\min[1,0] - P(H)} & \text{lainnya} \end{cases}$$

Rumus 2.1 Metode '*Net Belief*'

Dimana:

$CF(Rule)$ = faktor kepastian

$MB(H,E)$ = *measure of belief* (ukuran kepercayaan) terhadap hipotesis H, jika diberikan *evidence* E (antara 0 dan 1)

$MD(H,E)$ = *measure of disbelief* (ukuran ketidakpercayaan) terhadap *evidence* H jika diberikan *evidence* E (antara 0 dan 1)

$P(H)$ = probabilitas kebenaran H

$P(H|E)$ = probabilitas bahwa H benar karena fakta E

Contoh:

Seandainya seorang pakar penyakit kelamin menyatakan bahwa probabilitas seseorang berpenyakit phimosis adalah 0.02. dari data lapangan menunjukkan bahwa dari 100 orang penderita penyakit phimosis, 40 orang memiliki gejala kulup berminyak. Dengan menganggap H = Phimosis dan E = Kulup Berminyak, hitung faktor kepastian bahwa phimosos disebabkan oleh adanya kulup berminyak.

Jawab:

$$P(\text{Phimosis}) = 0.02$$

$$P(\text{Phimosis} \mid \text{Kulup Berminyak}) = 40/100 = 0.4$$

$$\begin{aligned} \text{MB}(H,E) &= \frac{\max[p(H|E), p(H)] - p(H)}{\max[1,0] - p(H)} \\ &= \frac{\max[0.4, 0.02] - 0.02}{1 - 0.02} = \frac{0.4 - 0.02}{1 - 0.02} = 0.39 \end{aligned}$$

$$\begin{aligned} \text{MD}(H,E) &= \frac{\min[p(H|E), p(H)] - p(H)}{\min[1,0] - p(H)} \\ &= \frac{\min[0.4, 0.02] - 0.02}{0 - 0.02} = \frac{0.02 - 0.02}{-0.02} = 0 \end{aligned}$$

$$\text{CF} = 0.39 - 0 = 0.39$$

Rule: IF (Gejala = Kulup Berminyak) THEN Penyakit = Phismosis (CF = 0.39)

2. Dengan cara mewawancarai seorang pakar

Nilai $\text{CF}(\text{Rule})$ didapat dari interpretasi “*term*” dari pakar, yang diubah menjadi nilai CF tertentu sesuai tabel berikut.

Contoh:

Pakar: Jika batuk dan panas, maka ‘hampir dipastikan’ (almost certainly) penyakitnya adalah influenza

Rule: IF (batuk AND panas) THEN penyakit = influenza (CF = 0.8)

Tabel 2.1 Tabel nilai CF

<i>Uncertain Term</i>	CF
<i>Definitely not</i> (pasti tidak)	-1.0
<i>Almost certainly not</i> (hampir pasti tidak)	-0.8
<i>Probably not</i> (kemungkinan besar tidak)	-0.6
<i>Maybe not</i> (mungkin tidak)	-0.4
<i>Unknown</i> (tidak tahu)	-0.2 to 0.2
<i>Maybe</i> (mungkin)	0.4
<i>Probably</i> (kemungkinan besar)	0.6
<i>Almost certainly</i> (hampir pasti)	0.8
<i>Definitely</i> (pasti)	1.0

Sumber: Sutojo, dkk. (2011: 195)

Menurut Sutojo, dkk. (2011: 196) secara umum, *rule* direpresentasikan dalam bentuk sebagai berikut.

IF E_1 AND E_2 AND E_N THEN H (CF rule)

Atau

IF E_1 OR E_2 OR E_N THEN H (CF rule)

Di mana:

$E_1 \dots E_N$: Fakta-fakta (*evidence*) yang ada

H : Hipotesis atau konklusi yang dihasilkan

CF Rule : Tingkat keyakinan terjadinya hipotesis H akibat adanya fakta-fakta $E_1 \dots E_N$

1. *Rule* dengan evidence E tunggal dan Hipotesis H tunggal.

IF E THEN H (CF rule)

$$CF(H,E) = CF(E) \times CF(rule)$$

Catatan: Secara praktik, nilai CF *rule* ditentukan oleh pakar, sedangkan nilai CF(E) ditentukan oleh pengguna saat berkonsultasi dengan sistem pakar.

2. Rule dengan evidence E ganda dan hipotesis H tunggal.

IF E_1 AND E_2 AND E_N THEN H (CF rule)

$$CF(H,E) = \min[CF(E_1), CF(E_2), \dots, CF(E_N)] \times CF(rule)$$

IF E_1 OR E_2 OR E_N THEN H (CF rule)

$$CF(H,E) = \max[CF(E_1), CF(E_2), \dots, CF(E_N)] \times CF(rule)$$

3. Kombinasi dua buah rule dengan evidence berbeda (E_1 dan E_2), tetapi hipotesis sama.

IF E_1 THEN H Rule 1 $CF(H,E_1) = CF_1 = C(E_1) \times CF(Rule1)$

IF E_2 THEN H Rule 2 $CF(H,E_2) = CF_2 = C(E_2) \times CF(Rule2)$

$$CF(CF_1, CF_2) = \begin{cases} CF_1 + CF_2 (1 - CF_1) & \text{jika } CF_1 > 0 \text{ dan } CF_2 > 0 \\ \frac{CF_1 + CF_2}{1 - \min[|CF_1|, |CF_2|]} & \text{jika } CF_1 < 0 \text{ atau } CF_2 < 0 \\ CF_1 + CF_2 \times (1 + CF_1) & \text{jika } CF_1 < 0 \text{ dan } CF_2 < 0 \end{cases}$$

Rumus 2.2 Kombinasi 2 *rule* dengan E berbeda

Menurut Sutojo, dkk. (2011: 204) kelebihan dan kekurangan *certainty factors* adalah:

a. Kelebihan

1. Metode ini cocok dipakai dalam sistem pakar yang mengandung ketidakpastian.
2. Dalam sekali proses perhitungan hanya dapat mengolah 2 data saja sehingga keakuratan data dapat terjaga.

b. Kelemahan

1. Pemodelan ketidakpastian yang menggunakan perhitungan metode certainty factors biasanya masih diperdebatkan.
2. Untuk data lebih dari 2 buah, harus dilakukan beberapa kali pengolahan data.

2.2 Variabel

Menurut Sudaryono (2015: 16) variabel penelitian adalah atribut atau sifat, atau nilai dari orang, objek, atau kegiatan yang mempunyai variasi tertentu yang ditetapkan oleh peneliti untuk dipelajari, kemudian ditarik kesimpulannya. Variabel adalah segala sesuatu yang memiliki variasi nilai. Contohnya, nilai ujian bervariasi dari 0-100; tingkat motivasi bisa bervariasi dari sangat rendah hingga sangat tinggi, dan lain sebagainya.

2.2.1 Kepribadian

Menurut Andreyana, dkk. (2015: 81) kepribadian adalah keseluruhan sikap, perasaan, ekspresi, tempramen, cirri-ciri khas dan perilaku seseorang. Sikap perasaan ekspresi dan tempramen ini akan terwujud dalam tindakan seseorang jika dihadapkan pada situasi tertentu. Setiap orang mempunyai kecenderungan perilaku yang baku, atau berlaku terus menerus secara konsisten dalam menghadapi situasi yang dihadapi, sehingga menjadi cirri khas pribadinya.

2.2.2 Tes *Myers-Briggs Type Indicator* (MBTI)

Menurut Zaman dan Abdillah (2009: 15) tes kepribadian MBTI (*Myers-Briggs Type Indicator*) adalah cara memetakan tipe kepribadian. Metode yang digunakan dalam tes kepribadian MBTI jauh lebih lengkap dan valid dibandingkan dengan metode lainnya. Metode MBTI membagi kepribadian manusia menjadi 16 tipe, sehingga di yakini memiliki tingkat akurasi yang lebih baik. Pengujian tes kepribadian MBTI berlaku untuk perusahaan dalam bentuk dan bidang apa pun. Tidak ada syarat khusus untuk sebuah perusahaan menggunakan tes kepribadian MBTI sebagai alat tes menguji dan mengevaluasi kepribadian pegawainya.

Sedangkankan menurut Dewantara (2014: 27) tes MBTI (*Myers-Briggs Type Indicator*) adalah tes psikotes yang bertujuan untuk mengetahui tipe-tipe kepribadian seseorang. Tes ini dikembangkan oleh Katherine Cook Briss dan puterinya Isabel Briggs. Mereka mengembangkan tes ini sejak Perang Dunia II

(1939-1945). Mereka percaya bahwa pengetahuan akan kepribadian dapat membantu perempuan yang akan memasuki dunia kerja di bidang industri. Setelah mengalami pengembangan, akhirnya Test MBTI ini pertama kali dipublikasikan pada tahun 1962.

Menurut Zaman dan Abdillah (2009: 20) berdasarkan metode MBTI, ada empat dimensi *preferensi* (kecenderungan) kepribadian manusia. Setiap dimensi menampilkan dua preferensi kepribadian yang berpasangan. Preferensi kepribadian ini didasarkan pada hal-hal berikut.

Tabel 2.2 Preferensi (kecenderungan) kepribadian

Preferensi	Kepribadian
Bagaimana memusatkan perhatian	<i>Extrovert - Introvert</i>
Bagaimana menerima informasi dari luar	<i>Sensing - Intuition</i>
Bagaimana menarik kesimpulan dan keputusan	<i>Thinking - Feelin</i>
Bagaimana pola hidup	<i>Judging - Perceiving</i>

Sumber: Zaman dan Abdillah (2009: 20)

Menurut Zaman dan Abdillah (2009: 40) terdapat 16 tipe kepribadian hasil kombinasi 4 dimensi *preferensi* (kecenderungan) yaitu:

Tabel 2.3 Enam belas tipe kepribadian

<i>Extrovert</i> (E)	← →	<i>Introvert</i> (I)	→	ESTJ	ENTJ	ESFJ	ENFJ
<i>Sensing</i> (S)	← →	<i>Intuition</i> (N)		ESTP	ESFP	ENTP	ENFP
<i>Thinking</i> (T)	← →	<i>Feeling</i> (F)		ISTJ	INTJ	ISFJ	INFJ
<i>Judging</i> (J)	← →	<i>Perceiving</i> (P)		ISTP	ISFP	INTP	INFP

Sumber: Zaman dan Abdillah (2009: 40)

2.2.2.1 Kepribadian *Extrovert* - *Introvert*

1. *Extrovert*

Menurut Zaman dan Abdillah (2009: 20) seorang *extrovert* dapat dengan mudah dan terbuka mengungkapkan masalah dan perasaan dalam dirinya kepada siapa pun yang dikenalnya. Pribadi yang tidak suka berdiam diri ini mengutamakan tindakan tanpa banyak merenungkan. Baginya, yang penting didahulukan adalah tindakan, baru memikirkan risiko yang akan menyimpannya. Orang dengan kepribadian ini juga memiliki pandangan hidup luas, sehingga mudah tertarik dengan tuntutan-tuntutan dari kondisi-kondisi di luar dirinya.

2. *Introvert*

Menurut Zaman dan Abdillah (2009: 21) seorang *introvert* cenderung menyimpan banyak rahasia tentang persoalan dirinya, juga banyak menjaga rahasia persoalan orang lain. Seorang *introvert* dikenal sebagai sosok pendiam dan sukar diduga, serta sering menarik diri dari suasana yang ramai. Seorang *introvert* juga jauh lebih memikirkan risiko yang akan terjadi sebelum mengambil tindakan.

2.2.2.2 Kepribadian *Sensing* - *Intuition*

1. *Sensing*

Menurut Zaman dan Abdillah (2009: 25) seorang *sensing* menilai bahwa apa yang dilihat, didengar, dicium dan diraba adalah dasar bagi dirinya

untuk mencari menanggapi, atau memahami informasi yang didapatnya. Baginya, fungsi indrawi menjadi alat ukur yang nyata dalam memandang situasi. Orang dengan kepribadian ini juga lebih melihat pada hal-hal yang fisik daripada metafisik. Dalam menganalisis masalah, ia akan menguraikan berdasarkan pengamatan pada peristiwa yang terjadi di lapangan dan selalu memerhatikan rambu-rambu atau tata tertib yang berlaku pada lingkungan pekerjaan. Seorang *sensing* juga sangat realistis dan cenderung tidak larut dalam pandangan-pandangan yang imajinatif.

2. *Intuition*

Menurut Zaman dan Abdillah (2009: 26) dalam mencermati informasi, seorang *intuition* cenderung menghubungkannya sesuatu yang dianggap memiliki keterkaitan atau bersifat korelatif. Ia tidak melihat apa yang terjadi, tetapi cenderung mencari fenomena apa yang menyebabkan hal tersebut terjadi. Sosok yang imajinatif ini bergairah dengan hal-hal yang abstrak, sehingga tidak heran jika ia sering disebut dengan pengkhayal. Dalam mengerjakan sesuatu, seorang *intuition* tidak mementingkan dari mana memulainya, yang terpenting baginya adalah melakukan terobosan-terobosan dengan mencari kesempatan-kesempatan untuk mendapatkan hal yang baru. Baginya, fungsi indrawi hanya media atau pintu untuk menyerap informasi, bukan untuk mempersepsi sebuah informasi. Pandangannya terhadap dunia muncul lewat proses penghayatan.

2.2.2.3 Kepribadian *Thinking - Feeling*

1. *Thinking*

Menurut Zaman dan Abdillah (2009: 30) seorang *thinking* mendasarkan keputusannya dengan pertimbangan logika dan nalar, tetapi kurang memedulikan perasaan orang lain. Orang dengan kepribadian ini sangat tegas dalam memutuskan dan memilih pekerjaan dengan alasan-alasan yang rasional. Sifat tegas ini tertanam karena ia termotivasi oleh kebutuhan prestasi dan pencapaian sasaran dalam menyelesaikan suatu pekerjaan atau ketika menyikapi situasi. Seorang dengan kepribadian ini juga apik melihat dan menemukan kesalahan. Dalam berdiskusi, sering menawarkan banyak solusi. Dalam membuat kesepakatan, ia pandai bernegosiasi. Selain itu, ia juga cermat dalam menghitung logika keuntungan dan kerugian, serta puas dengan hasil keputusannya sendiri.

2. *Feeling*

Menurut Zaman dan Abdillah (2009: 31) seorang *feeling* mengutamakan bekerja yang didasarkan pada keyakinan diri sendiri daripada kepada orang. Dalam memutuskan hal, seorang *feeling* sangat memerhatikan perasaan. Hal ini dilakukan demi menjaga hubungan baik dengan rekan-rekannya. Dalam mengambil keputusan maupun mengambil kesimpulan, ia cenderung subjektif. Sulit untuk mempertimbangkan dengan dasar benar atau salah (objektif). Oleh karenanya, terkadang ia plin-plan atau bahkan kebingungan dalam mengambil sikap.

2.2.2.4 Kepribadian *Judging* - *Perceiving*

1. *Judging*

Menurut Zaman dan Abdillah (2009: 34) seorang *judging* tidak suka dengan kejutan. Pola sikap yang ditunjukkan teratur. Ia senang dengan keputusan yang pasti. Dalam melakukan tindakan, ia terencana dengan jelas, serta cenderung menyukai kategori-kategori dan batasan-batasan yang tegas. Untuk mengatur kegiatannya, ia terkadang memiliki catatan atau jadwal kegiatan yang sudah direncanakan sebelumnya. Selain itu, seorang *judging* juga yang terkesan kurang luwes, tetapi berorientasi kepada aturan dan hasil saat mengerjakan sesuatu. Baginya, tugas yang diberikan adalah tanggung jawab yang harus diselesaikan. Namun, ia cenderung tidak bisa mengerjakan dua pekerjaan pada waktu yang sama. Ia baru akan mengerjakan tugas lain, jika tugas sebelumnya sudah dikerjakan dengan baik.

2. *Perceiving*

Menurut Zaman dan Abdillah (2009: 35) dalam menjalani hidup, seorang *perceiving* menyikapinya dengan terbuka, serta menerima kemungkinan-kemungkinan yang akan terjadi. Dengan begitu, ia mudah menyesuaikan diri memahami orang lain, dan mudah beradaptasi dengan lingkungan pekerjaan. Ia cenderung luwes karena memiliki rasa ingin tahu yang tinggi. Pada diri yang optimis ini, prestasi adalah prioritas dalam hidupnya, oleh karenanya ia berupaya untuk meraih keinginannya dalam situasi apa pun. Pada diri yang spontan ini, apa yang sudah dijadwalkan

hanya dijadikan sebagai pengingat saja. Keteraturan baginya adalah hal yang kaku. Ia lebih memilih bebas dan sangat menikmati proses yang sedang ia kerjakan (orientasi pada proses).

2.2.2.5 Karakteristik 16 Tipe Kepribadian MBTI

Menurut Zaman dan Abdillah (2009: 41) karakteristik ke-16 tipe kepribadian MBTI adalah sebagai berikut.

Tabel 2.4 Tipe Kepribadian ESTJ dan INFP

ESTJ (Pemikir yang <i>Extrovert</i> dengan Indera)	INFP (Perasa yang <i>Introvert</i> dengan Intuisi)
Kelebihan: • Praktis, realistis, berpegang pada fakta, dengan dorongan alamiah untuk bisnis dan mekanistik. • Bisa diandalkan, bertanggung jawab, dan taat pada ketentuan dan aturan. • Suka diberi wewenang untuk mengatur. • Mengorganisir proyek-proyek, lalu mengusahakan agar terselesaikan. • Bersifat logis, analitis, dan kritis secara objektif. • Fokus pada pekerjaan, bukan orang di balik pekerjaan. • Mengorganisasikan fakta-fakta, situasi dan operasi yang berhubungan dengan suatu proyek. • Membuat usaha sistematis untuk mencapai sasaran mereka pada waktunya. • Kepercayaan bahwa perilaku harus	Kelebihan: • Kesetiaan terhadap tugas dan kewajiban yang berhubungan dengan gagasan atau orang yang mereka pedulikan. • Mengambil pendekatan yang sangat pribadi terhadap hidup. • Menilai segala sesuatu dengan cita-cita batiniah dan nilai-nilai pribadi. • Berpegang teguh pada impian ideal mereka dengan keyakinan yang bersemangat. • Toleransi, fleksibilitas, dan kemampuan beradaptasi dalam masalah-masalah sehari-hari. • Penuh dengan antusiasme dan kesetiaan, tetapi jarang mengungkapkannya sampai mengenal betul orang yang diajak bicara. • Peduli pada proses belajar, ide, bahasa dan pekerjaan mandiri.

Tabel 2.4 Lanjutan

dikendalikan oleh logika. • Hidup dengan serangkaian aturan yang tertentu yang merangkum penilaian dasar mereka mengenai dunia.	• Cenderung untuk mengambil terlalu banyak, tetapi menyelesaikannya sebagian. • Bersahabat, tetapi sering terlibat pada apa yang dilakukannya. • Perhatian sedikit pada miliknya dan lingkungan sekitar.
Kelemahan: • Mengukur orang dengan standar atau kacamata dirinya. • Kurang memikirkan dampak terhadap orang lain. • Tergesa-gesa mengambil kesimpulan,	Kelemahan: • Sensitif terhadap kritikan. • Tidak sabar dengan hal detail dan rutin. • Menghindar dari konflik yang mungkin saja dinamis dan
tanpa dukungan informasi yang banyak. • Kurang peduli dengan pendapat orang lain. • Resisten dengan perubahan.	produktif. • Enggan melemparkan kritikan kepada orang lain. • Terkesan menyendiri dan kurang bersahabat.

Sumber: Zaman dan Abdillah (2009: 41)

Tabel 2.5 Tipe Kepribadian ENFP dan ISTJ

ENFP (Intuitif yang <i>Extrovert</i> dengan Perasaan)	ISTJ (Pengindera yang <i>Introvert</i> dengan Pikiran)
Kelebihan: • Mencari kemungkinan-kemungkinan dan cara-cara baru melakukan berbagai hal. • Memecahkan masalah-masalah sulit dengan cara-cara yang sederhana. • Mengubah proyek dan peluang-peluang baru demi kreativitas. • Kesempatan untuk mengembangkan dan mengilhami potensi dalam diri orang-orang lain. • Menciptakan lingkungan yang penuh semangat dan motivasi.	Kelebihan: • Dapat diandalkan, akurat. • Penghargaan yang lengkap, realistik dan praktis terhadap fakta. • Menerima tanggung jawab melampaui panggilan tugas. • Tampil tenang dan tegar dalam suatu krisis. • Mendalam, bekerja keras dan stabil. • Pilihan yang berhati-hati terhadap pekerjaan

Tabel 2.5 Lanjutan

• Mengkaji hal-hal dan terus mempertimbangkan pemecahannya. • Hangat, antusias tinggi, imajinatif dan pandai menemukan hal-hal baru. • Mampu melakukan apa pun yang menarik baginya. Cepat memberikan solusi dan cepat membantu orang yang punya masalah. • Sering mengandalkan kemampuannya untuk improvisasi daripada menyiapkannya dulu. • Sering menemukan alasan yang menarik untuk meyakinkan orang lain tentang apa yang diinginkannya.	• Kemampuan berorganisasi tinggi. • Serius, tenang, serta mencapai sukses dengan konsentrasi dan ketelitian. • Praktis, teratur, senang pada fakta, logis, realistis dan dapat diandalkan. • Melihat segala sesuatu dapat diorganisasikan dengan baik. • Bertanggung jawab. • Punya pendirian sendiri tentang apa yang harus dicapai dan mengerjakannya dengan mantap tidak peduli pada protes atau gangguan.
Kelemahan: • Karena kebebasannya, kurang patuh dengan aturan dan kurang disiplin dengan waktu. • Bekerja tidak terencana dan melompat-lompat, terkadang keluar jalur. • Banyak pekerjaan tapi hanya sedikit yang selesai. • Abai dengan fakta, data-data detail. • Lambata dan sulit mengambil keputusan dan kesimpulan karena terlalu banyak informasi. • Cenderung tidak logis karena mengutamakan menjaga perasaan semua pihak.	Kelemahan: • Kurang kreatif dan inovatif karena sangat fokus pada tugas. • Kurang peduli dengan kebutuhan orang lain, terutama dengan orang yang berbeda ide dengannya. • Ragu dengan hal-hal atau ide yang baru jika tidak melihat atau mengalaminya. • Dapat mengabaikan pendapat orang lain. • Jarang mengapresiasi orang lain.

Sumber: Zaman dan Abdillah (2009: 43)

Tabel 2.6 Tipe Kepribadian INTP dan ESFJ

INTP (Pemikir yang <i>Introvert</i> dengan Intuisi)	ESFJ (Perasa yang <i>Extrovert</i> dengan Indera)
Kelebihan: • Senang memecahkan masalah dengan logika dan analisis. • Biasanya tertarik pad aide-ide dengan hubungan sedikit pada diskusi. • Cenderung memiliki minat yang jelas. • Membuat penilaian logis tentang kemungkinan-kemungkinan non-personal. • Kritis secara objektif. • Lingkaran kecil sahabat-sahabat dekat. • Pendiam dan penyegan. • Sangat ingin tahu mengenai gagasan-gagasan baru. • Kerinduan untuk memahami misteri kompleks dari hal-hal yang bukan pribadi.	Kelebihan: • Hati yang hangat, banyak bicara, populer, punya hati nurani kuat, dilahirkan untuk bekerjasama dan anggota kelompok yang aktif. • Selalu melakukan sesuatu yang manis bagi orang lain. • Kerja dengan baik dalam situasi yang mendukung dan memujinya. • Minat utamanya ada dalam bidang-bidang yang secara langsung atau jelas. • Memancarkan simpati dan persekutuan. • Hubungan manusiawi yang harmonis. • Bersahabat, bijaksana dan simpatik. • Tekun, teliti dan teratur. • Berkonsentrasi pada sifat-sifat orang lain yang patut dipuji. • Belah kasihan dan kesadaran akan kondisi fisik.
Kelemahan: • Cermat dalam melihat kelemahan, tetapi segan mengutarakannya. • Bisa muncul sikap arogan, kurang komunikatif dan tidak mau mendengarkan orang lain. • Dapat melukai perasaan orang lain. • Enggan dengan pekerjaan yang detail, kurang dengan tindakan yang nyata.	Kelemahan: • Karena begitu tinggi perhatiannya terhadap orang, terkadang kurang memerhatikan diri sendiri. • Kesulitan meminta pertolongan, khawatir mengecewakan orang lain. • Susah melemparkan dan menerima kritik. • Keputusan atau kesimpulan terkadang subjektif dan tidak logis

Tabel 2.6 Lanjutan

	dengan pertimbangan rasa.
--	---------------------------

Sumber: Zaman dan Abdillah (2009: 44)

Tabel 2.7 Tipe Kepribadian ENTP dan ISFJ

ENTP (Intuitif yang <i>Extrovert</i> dengan Pikiran)	ISFJ (Pengindera <i>Introvert</i> dengan Perasaan)
Kelebihan: • Gesit, inovatif dan baik dalam banyak hal. • Menstimulasi orang lain, serta waspada dan banyak bicara. • Punya banyak cara untuk memecahkan masalah dan tantangan, tetapi bisa mengabaikan tugas-tugas rutin. • Cenderung melakukan hal baru yang menarik hati setelah melakukan sesuatu yang lain. • Terampil untuk menemukan alasan yang tepat tentang apa yang diinginkannya. • Mahir berurusan dengan teori dan hal-hal abstrak. • Mencari kemungkinan-kemungkinan dan cara-cara baru melakukan banyak hal. • Imajinasi dan inisiatif memulai pekerjaan. • Bertujuan untuk memahami daripada menghakimi orang. • Membangkitkan semangat dan efektif dalam memotivasi orang lain.	Kelebihan: • Tenang, ramah dan teliti. • Bekerja untuk memenuhi kewajiban. • Memberikan stabilitas dalam bekerja dan proyek. • Mendasarkan keputusan pada nilai-nilai pribadi. • Penghargaan yang lengkap, realistik dan praktis terhadap fakta. • Menerima tanggung jawab melampaui panggilan tugas. • Terampil, tenang dan tegar dalam suatu krisis. • Mendalam, bekerja keras dan stabil. • Kemampuan berorganisasi tinggi. • Ramah, simpatik, banyak akal dan terbebani dengan tulus.
Kelemahan: • Tidak cermat dengan hal-hal yang detail. • Kurang praktis dalam menyikapi hal yang sederhana.	Kelemahan: • Tidak fleksibel dengan perubahan situasi di tengah jalan. • Kurang berani menyampaikan pendapat.

Tabel 2.7 Lanjutan

• Tidak suka diberi tahu apa yang harus dilakukan. • Tidak betah dengan hal yang rutin dan detail. • Dapat menjadi arogan karena merasa banyak tahu.	• Tidak senang dengan obrolan abstrak dan imajinatif. • Sensitif dengan kritik. • Banyak pertimbangan, tidak spontan.
--	---

Sumber: Zaman dan Abdillah (2009: 45)

Tabel 2.8 Tipe Kepribadian INFJ dan ESTP

INFJ (Intuitif yang <i>Introvert</i> dengan Perasaan)	ESTP (Pengindera yang <i>Extrovert</i> dengan Pikiran)
Kelebihan: • Sukses karena ketekunan, originalitas dan keinginan kuat untuk melakukan apa saja yang diperlukan. • Memberikan yang terbaik dalam pekerjaan. • Dihormati karena keteguhan hati pada prinsip. • Biasanya diikuti dan dihormati karena kejelasan visi, serta dedikasi pada hal-hal baik. • Intuisi dan pemahaman. • Kesempatan untuk bersikap inovatif dengan gagasan baru. • Keharmonisan dan persekutuan. • Mahir membujuk orang-orang lain untuk menyetujui dan bekerja sama. • Subjektif dalam menimbang dan menerapkan nilai-nilai. • Sensitif dan peka, serta percaya pada keyakinan dan nilai pribadi.	Kelebihan: • Cenderung menyukai sesuatu yang mekanistik dan olahraga, dengan teman berada di sampingnya. • Mudah beradaptasi, toleran dan pada umumnya konservatif tentang nilai-nilai. • Tidak suka penjelasan terlalu panjang. • Paling baik dalam hal-hal nyata yang dapat dilakukan. • Menghadapi hidup secara realistic dan tidak pribadi. • Bersandar pada apa yang dilihat, dengar dan ketahui dari tangan pertama. • Mencari solusi-solusi yang memuaskan. • Kemampuan beradaptasi dengan pikiran terbuka dan sikap toleran. • Keterampilan pemecahan masalah. • Membuat keputusan berdasarkan analisis logis pikiran, bukan pada nilai-nilai perasaan pribadi.

Tabel 2.8 Lanjutan

Kelemahan: • Takut dan menjauhi konflik. • Tidak percaya pada informasi yang belum terbukti. Dapat terlalu memaksa membuat orang tidak senang. • Terlalu mengatur dan perfeksionis. • Mengabaikan bukti atau fakta yang tidak mendukung pendapatnya.	Kelemahan: • Tidak memiliki prioritas. • Pemecahan masalah tidak sistematis. • Kurang peduli dengan perasaan orang lain. • Tidak memiliki rencana yang matang, sesuatu disiapkan secara mendadak. • Kurang menganggap penting persoalan yang detail.
--	---

Sumber: Zaman dan Abdillah (2009: 47)

Tabel 2.9 Tipe Kepribadian ENFJ dan ISTP

ENFJ (Perasaan yang Extrovert dengan Perasaan)	ISTP (Pemikir yang Introvert dengan Indera)
Kelebihan: • Responsif dan bertanggung jawab. • Pada umumnya peduli dengan perkataan atau keinginan orang lain. • Cenderung melakukan sesuatu dengan memerhatikan perasaan orang lain. • Memimpin diskusi dengan cepat dan taktis. • Pandai bergaul, populer dan simpatik. • Responsif pada kritik dan pujian. • Hubungan manusiawi yang harmonis, bersahabat, bijaksana dan simpatik. • Tekun, teliti dan teratur. • Mengidealkan apa yang dikagumi. • Kemampuan melihat nilai dari pendapat orang lain. • Mudah setuju terhadap pendapat orang lain, dalam batas-batas yang	Kelebihan: • Biasanya tertarik dengan hubungan sebab akibat, bagaimana dan mengapa sesuatu terjadi. • Ketika mengorganisir sesuatu biasanya menggunakan prinsip-prinsip logis. • Kesenangan mengorganisir fakta dan data. • Biasanya <i>easy-going</i>, menyukai kesenangan dan mudah beradaptasi. • Kemampuan berorganisasi tinggi. • Keingintahuan besar, tetapi secara diam-diam. • Mudah beradaptasi, kecuali jika salah satu prinsip penting dilanggar. • Terampil. • Menyukai segala sesuatu yang

Tabel 2.9 Lanjutan

masuk akal.	memberikan banyak informasi bagi indera. • Tertarik pada bagaimana dan mengapa hal-hal bekerja.
Kelemahan: • Menanggapi kritik terlalu berlebihan. • Demi menghindari konflik, hubungan dengan orang lain terkadang kurang jujur. • Terlerna dengan misi pribadi demi menjunjung keselarasan. • Empati yang berlebihan, dapat mencampuri urusan pribadi orang lain. • Lupa dengan fakta yang penting.	Kelemahan: • Menyikapi persoalan terkadang memulainya tidak dari awal (dari tengah). • Lebih tertarik menampung pilihan yang ada, sehingga menjadi ragu. • Sulit memperlihatkan reaksi emosi, perasaan dan perhatiannya karena di anggap tidak penting. • Kurang ada tindak lanjut sampai selesai. • Kurang komunikasi.

Sumber: Zaman dan Abdillah (2009: 48)

Tabel 2.10 Tipe Kepribadian ESFP dan INTJ

ESFP (Pengindera <i>Extrovert</i> dengan Perasaan)	INTJ (Intuitif <i>Introvert</i> dengan Pikiran)
Kelebihan: • <i>Outgoing, easygoing</i>, mudah berteman dan bersahabat. • Mengetahui apa yang terjadi di sekelilingnya dan ikut serta dalam kegiatan tersebut. • Lebih cepat ingat pada kejadian-kejadian daripada teori-teori. • Bersandar pada apa yang mereka lihat, dengar dan ketahui dari tangan pertama. • Menerima dan menggunakan fakta di sekitar mereka. • Mencari solusi-solusi yang memuaskan.	Kelebihan: • Biasanya memiliki ide-ide original dan dorongan kuat untuk mencapai idenya dan tujuan-tujuannya. • Di bidang yang cocok, memiliki kekuatan untuk mengorganisir pekerjaan dan melakukannya dengan atau tanpa bantuan. • Kritis, mandiri dan lebih suka melakukan sesuatu dengan caranya sendiri. • Pikiran-pikiran dan juga tindakan-tindakannya inovatif. • Mempercayai pemahaman intuitif terhadap hubungan sejati dan makna dari hal-hal.

Tabel 2.10 Lanjutan

• Kemampuan beradaptasi tinggi. • Berpikiran terbuka dan toleransi. • Menggunakan aturan-aturan, sistem atau situasi pada saat ini sebagai penolong, bukan penghalang. • Membuat keputusan berdasarkan nilai-nilai perasaan pribadi, bukan berdasarkan analisis logis.	• Kerinduan untuk menghabiskan waktu dan berusaha untuk melihat inspirasi mereka dilaksanakan dalam praktek. • Mendorong orang lain hampir sekeras mereka mendorong diri sendiri. • Menganalisis berbagai alternative pemecahan masalah.
Kelemahan: • Menggampangkan menyikapi situasi, karenanya terkadang terlihat santai. • Tidak suka diberi petunjuk bagaimana seharusnya bekerja, dapat bermasalah dengan aturan dan batasan. • Karena mudah tergoda dan sulit mendisiplinkan diri. • Terkadang mengorbankan tugas dan membebani orang lain karena menempatkan prioritas pada pengalaman dan mencoba menikmati hidup.	Kelemahan: • Terkadang keras kepala. • Dapat menjadi skeptis yang berlebihan. • Terkadang arogan. • Kurang melibatkan orang lain dalam kegiatannya. • Terkadang kurang memedulikan standar orang lain.

Sumber: Zaman dan Abdillah (2009: 49)

Tabel 2.11 Tipe Kepribadian ENTJ dan ISFP

ENTJ (Pemikir yang <i>Extrovert</i> dengan Intuisi)	ISFP (Perasa yang <i>Introvert</i> dengan Pengindera)
Kelebihan: • Jujur dan terus terang, kuat kemauannya dan dapat menjadi pemimpin dalam kegiatan-kegiatan. • Biasanya cepat mendapat informasi dan menikmati informasi tersebut karena kesukaannya membaca. • Tindakan eksekutif dan perencanaan jangka panjang.	Kelebihan: • Sensitive, ramah, tidak menonjolkan diri, rendah hati pada kemampuannya. • Menghindari konflik, tidak memaksakan pendapat atau nilai-nilainya pada orang lain. • Biasanya tidak mau memimpin, tetapi menjadi pengikut yang setia.

Tabel 2.11 Lanjutan

• Bersifat logis, analitis dan kritis secara objektif. • Bersandar pada pemikiran. • Fokus pada gagasan, bukan orang di balik gagasan. • Berpikir lebih dulu, mengatur perencanaan, situasi dan operasi yang berhubungan dengan suatu proyek. • Membuat usaha sistematis untuk mencapai sasaran mereka pada waktunya. • Kepercayaan bahwa perilaku harus dikendalikan oleh logika. • Hidup dengan serangkaian aturan yang tertentu yang merangkum penilaian dasar mereka mengenai dunia. • Memandang apa yang tidak logis dan tidak konsisten.	• Seringkali santai menyelesaikan sesuatu karena sangat menikmati apa yang terjadi saat ini. • Menilai hidup dengan impian ideal batiniah dan nilai-nilai pribadinya. • Setia kepada tugas, dapat diandalkan dan akurat. • Toleran, berpikiran terbuka, fleksibel dan mampu beradaptasi. • Tampil tenang dan tegar dalam suatu krisis. • Ketekunan yang menstabilkan. • Ramah, simpatik, banyak akal dan terbeban dengan tulus. • Perfeksionis jika menyukai sesuatu secara mendalam.
Kelemahan: • Kurang kesabaran terhadap situasi yang tidak efisien. • Dingin dengan perasaan orang lain. • Tegas dengan aturan dan mengabaikan sisi manusiawi. • Terlalu percaya diri dan fokus pada dirinya sendiri. • Tampak arogan dengan pengetahuannya.	Kelemahan: • Terlalu santai menyelesaikan pekerjaan. • Kurang mampu menampilkan diri. • Kepemimpinannya kurang. • Karena hidupnya sepenuhnya untuk hari ini, cenderung tidak menyiapkan visi jangka panjang. • Sensitif dengan kritik, sehingga mudah terluka dan kecil hati.

Sumber: Zaman dan Abdillah (2009: 51)

2.3 Software Pendukung

2.3.1 Data Flow Diagram (DFD)

Menurut Rosa dan Shalahuddin (2016: 70) *Data Flow Diagram* atau dalam bahasa Indonesia menjadi Diagram Alir Data (DAD) 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. Oleh karena itu, DFD lebih sesuai digunakan untuk memodelkan fungsi-fungsi perangkat lunak yang akan diimplementasikan menggunakan pemrograman terstruktur karena pemrograman terstruktur membagi-bagi bagiannya dengan fungsi-fungsi dan prosedur-prosedur. DFD tidak sesuai untuk memodelkan sistem perangkat lunak yang akan dibangun menggunakan pemrograman berorientasi objek.

2.3.2 Unified Modeling Language (UML)

Menurut Rosa dan Shalahuddin (2016: 137) pada perkembangan teknik pemrograman berorientasi objek, muncul sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik

pemrograman berorientasi objek, yaitu *Unified Modeling Language* (UML). UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun dan dokumentasi dari sistem perangkat lunak. UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. UML hanya berfungsi untuk melakukan pemodelan. Jadi penggunaan UML tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya UML paling banyak digunakan pada metodologi berorientasi objek.

2.3.2.1 Use Case Diagram

Menurut Rosa dan Shalahuddin (2016: 155) *Use case* atau diagram *use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut.

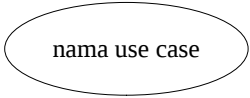
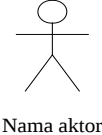
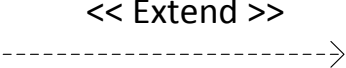
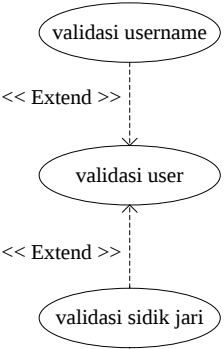
Menurut Rosa dan Shalahuddin (2016: 155) syarat penamaan pada *use case* adalah nama didefinisikan sesimpel mungkin dan dapat dipahami. Ada dua hal utama pada *use case* yaitu pendefinisian apa yang disebut aktor dan *use case*.

1. Aktor merupakan orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.

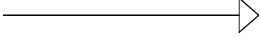
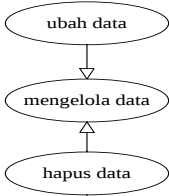
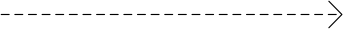
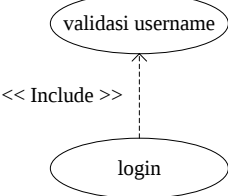
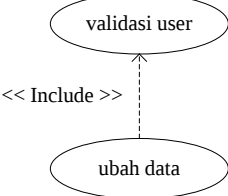
2. *Use case* merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.

Berikut adalah symbol-simbol yang ada pada diagram *use case*:

Tabel 2.12 Simbol-simbol pada *diagram use case*

Simbol	Deskripsi
<i>Use case</i> 	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>use case</i>.
Aktor / actor 	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.
Asosiasi / association 	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor
Ekstensi / extend 	Relasi <i>use case</i> tambahan kesebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu, biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan, misalnya:  arah panah mengarah pada <i>use case</i> yang ditambahkan, biasanya <i>use case</i> yang menjadi <i>extend</i>-nya merupakan jenis yang sama dengan <i>use case</i> yang menjadi induknya.

Tabel 2.12 Lanjutan

Generalisasi / <i>generalization</i> 	Hubungan generalisasi dan spesialisasi (umum - khusus) antara dua buah use case dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya, misalnya:  arah panah mengarah pada <i>use case</i> yang menjadi generalisasainya (umum).
Menggunakan / <i>include / uses</i> << Include >>  << Uses >> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini. Ada dua sudut pandang yang cukup besar mengenai <i>include</i> di <i>use case</i>: 1. <i>Include</i> berarti <i>use case</i> yang ditambahkan akan selalu dipanggil saat <i>use case</i> tambahan dijalankan, misal pada kasus berikut:
	 2. <i>Include</i> berarti <i>use case</i> yang ditambahkan akan selalu melakukan pengecekan apakah <i>use case</i> yang ditambahkan telah dijalankan sebelum <i>use case</i> tambahan dijalankan, misal pada kasus berikut:  Kedua sudut pandang diatas dapat dianut salah satu atau keduanya tergantung pada pertimbangan dan kebutuhan.

Sumber: Rosa dan Shalahuddin (2016: 155)

2.3.2.2 Activity Diagram

Menurut Rosa dan Shalahuddin (2016: 137) diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

Diagram aktivitas juga banyak digunakan untuk mendefinisikan hal-hal berikut:

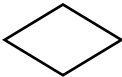
1. Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.
2. Urutan atau pengelompokan tampilan dari sistem/*user interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan.
3. Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya.
4. Rancangan menu yang ditampilkan pada perangkat lunak.

Berikut adalah simbol-simbol yang ada pada diagram aktivitas:

Tabel 2.13 Simbol-simbol pada *activity diagram*

Simbol	Deskripsi
Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal
Aktivitas	Aktivitas yang dilakukan sistem,

Tabel 2.14 Lanjutan

	aktivitas yang biasanya diawali dengan kata kerja
Percabangan / <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
Penggabungan / <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu

Sumber: Rosa dan Shalahuddin (2016: 137)

2.3.2.3 *Class Diagram*

Menurut Rosa dan Shalahuddin (2016: 141) diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

1. Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas
2. Operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas.

Diagram kelas dibuat agar pembuat program atau *programmer* membuat kelas-kelas sesuai rancangan di dalam diagram kelas agar dokumentasi perancangan dan perangkat lunak sinkron. Kelas-kelas yang ada pada struktur sistem harus dapat melakukan fungsi-fungsi sesuai dengan kebutuhan sistem. Susunan struktur kelas yang baik pada diagram kelas sebaiknya memiliki jenis-jenis kelas berikut:

1. Kelas main

Kelas yang memiliki fungsi awal dieksekusi ketika sistem dijalankan

2. Kelas yang menangani tampilan sistem (*View*)

Kelas yang mendefinisikan dan mengatur tampilan ke pemakai

3. Kelas yang diambil dari pendefinisian *use case* (*controller*)

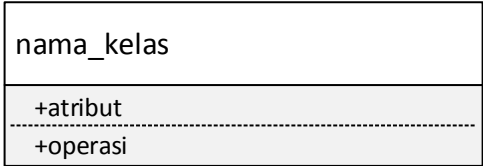
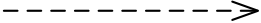
Kelas yang menangani fungsi-fungsi yang harus ada diambil dari pendefinisian *use case*, kelas ini biasanya disebut dengan kelas proses yang menangani proses bisnis pada perangkat lunak.

4. Kelas yang diambil dari pendefinisian data (*model*)

Kelas yang digunakan untuk memegang atau membungkus data menjadi sebuah kesatuan yang diambil maupun akan disimpan ke basis data.

Berikut adalah simbol-simbol yang ada pada diagram kelas:

Tabel 2.14 Simbol-simbol pada *class diagram*

Simbol	Deskripsi
Kelas 	Kelas pada struktur sistem
Antarmuka / <i>interface</i>  nama_interface	Sama dengan konsep interface dalam pemrograman berorientasi objek
Asosiasi berarah / <i>direct association</i> 	Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
Generalisasi	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum-khusus)
Kebergantungan / <i>dependency</i> 	Relasi antar kelas dengan makna kebergantungan kelas

Tabel 2.14 Lanjutan

Agresi / aggregation 	Relasi antar kelas dengan makna semua-bagian (<i>whole-part</i>)
---	--

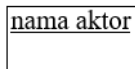
Sumber: Rosa dan Shalahuddin (2016: 146)

2.3.2.4 Sequence Diagram

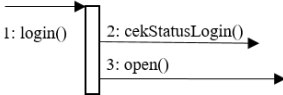
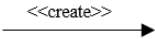
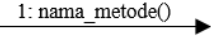
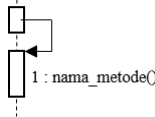
Menurut Rosa dan Shalahuddin (2016: 165) diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambar diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Membuat diagram sekuen juga dibutuhkan untuk melihat scenario yang ada pada *use case*.

Menurut Rosa dan Shalahuddin (2016: 165) simbol-simbol yang ada pada diagram sekuen adalah sebagai berikut.

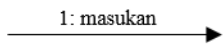
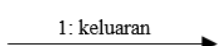
Tabel 2.15 Simbol-simbol pada *sequence diagram*

Simbol	Deskripsi
Aktor  nama aktor atau  nama aktor tanpa waktu aktif	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda diawal frase nama aktor

Tabel 2.15 Lanjutan

Garis hidup / <i>lifeline</i> 	Menyatakan kehidupan suatu objek
Objek nama objek : nama kelas	Menyatakan objek yang berinteraksi pesan
Waktu aktif 	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya, misalnya  Maka cekStatusLogin() dan open() dilakukan didalam metode login() Aktor tidak memiliki waktu aktif
Pesan tipe <i>create</i> 	Menyatakan suatu objek membuat objek lain, arah panah mengarah pada objek yang dibuat
Pesan tipe <i>call</i> 	sendiri,  Arah panah mengarah pada objek yang memiliki operasi/metode, karena ini memanggil operasi/metode maka operasi/metode yang dipanggil harus ada pada diagram kelas sesuai dengan kelas objek yang berinteraksi

Tabel 2.15 Lanjutan

Pesan tipe <i>send</i> 	Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim
Pesan tipe <i>return</i> 	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian
Pesan tipe <i>destroy</i> 	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada <i>create</i> maka ada <i>destroy</i>

Sumber: Rosa dan Shalahuddin (2016: 165)

2.3.3 Microsoft Visual Studio 2008

Menurut Wahana Komputer (2008: 2) keunggulan dari *Visual Studio* adalah pada *interface* desainnya yang menggunakan IDE (*Integrated Development Environment*). IDE *Visual Studio* sudah terintegrasi dengan kompiler, *.Net*, fasilitas *debug*, fasilitas pendistribusian *file* program yang telah jadi, fasilitas intelisense (*autocomplete modern*) dan lain sebagainya. Beberapa kelebihan dari *Visual Studio* 2008 adalah sebagai berikut.

1. *Integrated Development Environment* (IDE)

Pada IDE *Visual Studio* 2008 ditambahkan beberapa fitur diantaranya:

a. *Transfer Setting*

Jika sebuah komputer terinstalasi *Visual Studio 2005* dan *Visual Studio 2008 Express*, maka dapat dilakukan *transfer setting* dari *Visual Studio 2005 Express* ke *Visual Studio 2008 Express*.

b. *Community Components*

Ketika menciptakan *community component* menggunakan *Visual Studio 2008*, maka kita dapat memilih apakah component tersebut akan digunakan di versi *Visual Studio 2005* dan *Visual Studio 2008* atau hanya *Visual Studio 2008*.

c. *Community* dan menu *help*

Visual Studio Express 2008 memiliki sebuah menu baru yang terdapat dalam menu *help*, yaitu *MSDN Forums*. Menu tersebut menggantikan menu *community* yang diantaranya adalah menu *ask a question* dan *check question status*.

d. Manajemen *Windows* dan elemen-elemen umum unsur.

Beberapa elemen *user interface* telah diperbarui, yaitu:

- Elemen navigasi pada IDE: *interface* yang telah diperbarui dan ditingkatkan kemampuannya bisa membantu mempermudah perpindahan antara komponen IDE.
- *Tools windows docking* telah ditingkatkan kemampuannya sehingga memudahkan dilakukannya *docking windows*.
- *Common dialog box*: *Visual Studio 2008* menggunakan *dialog box* standar ketimbang *custom dialog box*.

- Dapat melakukan *setting font* untuk elemen-elemen pada IDE yang dapat diatur sendiri melalui menu *environment font*.

e. *Class designer* yang mendukung kode *Visual C++*

Pada *Visual Studio* sebelumnya, *class designer* hanya mendukung *C#* dan *visual basic*. Pada *Visual Studio 2008*, *class designer* mendukung *C++*, tetapi masih terbatas hanya dalam visualisasi dan dokumentasi saja.

f. Pengembangan program aplikasi *web*

Dalam *Visual Studio 2008* dapat mengembangkan program aplikasi *web* dengan mudah dan bisa memanfaatkan semua fitur dalam *Visual Studio 2003* dan *2005*. Penggunaan *AJAX* juga didukung secara penuh. Selain itu, update *web* bisa dilakukan per bagian dari sebuah halaman.

2. Proyek Desainer

Pada bagian *project designer*, *Visual Studio* memiliki beberapa fitur, yaitu:

a. Dukungan terhadap *WPF (Windows Presentation Foundation)*

Pada *Visual Studio 2008* terdapat penambahan program aplikasi *Windows Presentation Foundation*. Ada beberapa macam tipe aplikasi yang dikembangkan, yaitu:

- *WPF Application (.xaml, .exe)*
- *WPF Browser Application (.exe, .xbap)*
- *WPF Custom Control Library (.dll)*

- WPF *User Control Library* (.dll)

b. Dukungan *multitargeting*

Fasilitas *multitargeting* ini mengakibatkan developer bisa menuliskan kode untuk framework .net tertentu.

- .Net framework 2.0 termasuk dalam *Visual Studio* 2005.
- .Net framework 3.0 termasuk dalam *Windows Vista*.
- .Net framework 3.5 termasuk dalam *Visual Studio* 2008

3. *Deployment*

Aplikasi yang telah selesai dikerjakan untuk kemudian didistribusikan kepada *end user*, *Visual Studio* memberikan beberapa pilihan, yaitu:

a. *ClickOnce Deployment*

ClickOnce menyediakan beberapa fasilitas baru, antara lain:

- Dukungan terhadap WPF *Web Application*.
- Dukungan terhadap pembuatan *manifest* dibawah kontrol *user account* pada *Windows Vista*.
- Dukungan terhadap *deployment tools* dan *add-in* untuk aplikasi *office*.
- Dukungan terhadap *browser* pihak ketiga.
- Dukungan mengasosiasikan sebuah ekstensi file tertentu dengan aplikasi *ClickOnce*.
- Dapat melakukan perubahan lokasi *deployment*.

- Aplikasi *ClickOnce* selalu diinstalasi dan dijalankan berdasarkan *user* masing-masing sesuai dengan *account* sewaktu instalasi.

b. *Deployment* menggunakan *windows installer*

Deployment program aplikasi menggunakan *windows installer* telah di *update* untuk *Windows Vista* dan *.Net framework* yang lebih baru.

4. *Editing*

Pada sisi *editing* terdapat beberapa fasilitas baru, diantaranya:

a. *Design view* baru dan *design tools* untuk *CSS*

Pada *Visual Studio 2008* terdapat fasilitas *CSS (Cascading Style Sheet) editing* yang kaya akan *tool* baru.

b. *Intellisense* untuk *Jscript* dan *ASP .Net AJAX*

Visual Studio 2008 menyertakan fasilitas *intellisense* untuk *editing Jscript* dan *ASP .NET AJAX*.

c. *WPF Designer*

Visual Studio 2008 sudah memiliki fasilitas yang memungkinkan *developer* untuk mengembangkan aplikasi *WPF* dan *custom control* pada *IDE*.

5. *Pengolahan Data*

Dari segi pengolahan data, *Visual Studio* memiliki beberapa fasilitas baru, yaitu:

- a. Adanya *Object Relational Designer* (O/R Designer), yang akan membantu *developer* membuat dan melakukan *editing* obyek *LINQ to SQL* yang terletak di antara program aplikasi dan *database*.
 - b. Dukungan terhadap pemrograman multitier (N-tier) peningkatan terhadap *dataset designer* yang akan membantu *developer* dalam memisahkan kode *TableAdapter* dan kode *typed datasets* ke dalam proyek yang berbeda.
6. LINQ (*Language-Integrated Query*)

LINQ merupakan fasilitas terbaru dalam *Visual Studio* 2008, yang menunjang kemampuan *query* ke dalam sintaks C# dan *Visual Basic*. LINQ membuat seolah-olah *query* juga merupakan sintaks dari bahasa C# dan *Visual Basic*.

2.3.4 Bahasa Pemrograman *Visual C#*

Menurut Wahana Komputer (2015: 7) C# adalah sebuah bahasa pemrograman yang berorientasi pada objek yang dikembangkan oleh *Microsoft* dan menjadi salah satu bahasa pemrograman yang mendukung *.Net programming* melalui *Visual Studio*. C# didasarkan pada bahasa pemrograman C++, C# juga memiliki kemiripan dengan beberapa bahasa pemrograman seperti *Visual Basic*, *Java*, *Delphi* dan tentu saja C++. C# memiliki kemudahan *syntax* seperti *Visual Basic*, dan tentu saja ketangguhan seperti *Java* dan C++. C# bergantung pada

CLR yang juga merupakan sumber *library* bagi program *.Net* lain. Semua program C# memerlukan CLR untuk dapat dijalankan.

Menurut Wahana Komputer (2015: 13) ada beberapa alasan yang mendasari pemilihan bahasa C# untuk mengembangkan aplikasi, yaitu:

1. C# benar-benar *Object Oriented*

C# adalah bahasa yang benar-benar *object oriented*. Ini dapat dilihat dari kemampuan C# dalam pembentukan objek, *class*, melakukan *encapsulation*, *inheritance* dan *polymorphism* dengan mudah.

2. C# sangat sederhana

Bahasa C# bersifat sederhana karena didasarkan pada bahasa C dan C++ bahkan bahasa *Java*. Akan tetapi, C# lebih sederhana dari bahasa-bahasa tersebut karena C# dibuat dengan menghilangkan kelemahan-kelemahan dari bahasa-bahasa yang mendasarinya.

3. C# merupakan bahasa pemrograman *modern*

Bahasa C# memiliki fasilitas *exception handling*, *garbage collection*, *extensible data types*, *secure code*. Fasilitas-fasilitas ini yang akan ada di tiap bahasa pemrograman yang *modern*. C# sudah memilikinya.

4. Mampu membuat berbagai aplikasi

Dengan C#, kita dapat membuat berbagai macam aplikasi, mulai dari aplikasi *console*, *pengolah kata*, *form web* dan lain-lain.

5. Efisien

Bahasa C# merupakan bahasa yang mengandalkan *library* yang sangat lengkap. Oleh karena itu, bahasa C# hanya memiliki sedikit keywords.

Jadi, para pengembang dapat mengingatnya dan memahami kegunaannya dengan baik.

6. C# bersifat modular

Bahasa C# sangat modular, tiap class disimpan dalam namespace yang dapat dimanfaatkan kembali oleh program lain yang membutuhkannya.

7. Bahasa pemrograman masa depan

C# akan menjadi bahasa msa depan karena dukungan penuh Microsoft terhadap teknologi *.Net*. Bahasa ini adalah salah satu bahasa pemrograman terbaik yang dibuat dengan didasarkan pada 4 bahasa lain yang telah ada, yaitu VB, C++, C, *Java*, tetapi tidak mewarisi kelemahan masing-masing bahasa tersebut.

2.3.5 SQL Server 2008

Menurut Wahana Komputer (2010: 2), SQL Server 2008 adalah sebuah terobosan baru dari *Microsoft* dalam bidang *database*. SQL Server adalah sebuah DBMS (*Database Management System*) yang dibuat oleh *Microsoft* untuk ikut berkecimpung dalam persaingan dunia pengolahan data. *Microsoft* merilis SQL Server 2008 dalam beberapa versi yang disesuaikan dengan segment-segment pasar yang dituju. Versi tersebut adalah sebagai berikut.

1. Versi *Compact*

Ini adalah versi “Tipis” dari semua versi yang ada. Versi ini seperti versi *desktop* pada SQL Server 2000. Versi ini juga digunakan pada *handheld device* seperti *Pocket PC*, *PDA*, *Smart Phone* dan *Tablet PC*.

2. Versi *Express*

Ini adalah versi “Ringan” dari semua versi yang ada (tetapi versi ini berbeda dxengan versi *compact*) dan paling cocok untuk latihan para pengembang aplikasi. Versi ini memuat *Express Manager* standar, integrasi dengan CLR dan XML.

3. Versi *Workgroup (Workgroup Edition)*

Versi ini dirancang untuk kalangan bisnis berskala kecil dan biasanya digunakan pada level dalam departemen saja. Versi ini menyediakan dukungan terhadap relasional *database*, tetapi tanpa fasilitas *Business Intelligent*. Versi atau edisi ini mendukung 2 prosesor dan memori sebesar 2GB.

4. Versi standar (*Standard Edition*)

Versi ini menyediakan apa yang dimiliki oleh versi *workgroup*, tetapi versi ini sudah mendukung 4 prosesor dan menyediakan versi 32 bit dan 64 bit. Versi ini juga menyertakan fungsi *Business Intelligent*.

5. Versi *Enterprise (Enterprise Edition)*

Versi ini memiliki semua fasilitas yang ada pada versi standar, tetapi versi ini mampu mengangani user yang banyak. Fasilitas lain yang

diberikan adalah manajemen *database* secara *online*, *data partitioning* dan *database snapshot*.

6. Versi *Developer* (*Developer Edition*)

Versi ini memiliki semua keunggulan dari versi *enterprise*. Versi ini juga memiliki versi 32 bit dan 64 bit. Hal yang khusus pada versi ini adalah lisensi (izin penggunaan) yang diberikan hanya untuk pengembangan, *testing* dan demonstrasi aplikasi.

2.3.6 Pengujian Perangkat Lunak

Menurut Rosa dan Shalahuddin (2016: 271) sering perangkat lunak mengandung kesalahan (*error*) pada proses-proses tertentu pada saat perangkat lunak sudah berada di tangan *user*. Kesalahan-kesalahan (*error*) pada perangkat lunak ini sering disebut dengan “*bug*”. Untuk menghindari *bug* maka diperlukan adanya pengujian perangkat lunak sebelum perangkat lunak diberikan ke pelanggan atau selama perangkat lunak masih terus dikembangkan.

Pengujian diperlukan tidak hanya untuk meminimalisasi kesalahan secara teknis tapi juga kesalahan non teknis (misalnya pengujian pesan kesalahan sehingga *user* tidak bingung atau tidak mengerti dengan pesan kesalahan yang muncul, atau juga jika masukan dan keluaran yang diperlukan berkapasitas sangat besar). Pengujian adalah satu set aktifitas yang direncanakan dan sistematis untuk menguji atau mengevaluasi kebenaran yang diinginkan. Aktifitas pengujian terdiri dari satu set atau sekumpulan langkah dimana dapat menempatkan desain kasus

uji yang spesifik dan metode pengujian. Secara umum pola pengujian pada perangkat lunak sebagai berikut:

1. Pengujian dimulai dari level komponen hingga integrasi antar komponen menjadi sebuah sistem.
2. Teknik pengujian berbeda-beda sesuai dengan berbagai sisi atau unit uji dalam waktu yang berbeda-beda pula bergantung pada pengujian pada bagian mana yang dibutuhkan.
3. Pengujian dilakukan oleh pengembang perangkat lunak, dan jika untuk proyek besar, pengujian bisa dilakukan oleh tim uji yang tidak terkait dengan tim pengembang perangkat lunak (*Independent Test Group* (ITG)).
4. Pengujian dan penirkutuan (*debugging*) merupakan aktifitas yang berbeda, tetapi penirkutuan (*debugging*) harus diakomodasi pada berbagai strategi pengujian. Pengujian lebih fokus untuk mencari adanya kesalahan (*error*) baik dari sudut pandang orang secara umum atau dari sudut pandang pengembang tanpa harus menemukan lokasi kesalahan pada kode program. Penirkutuan (*debugging*) adalah proses mencari lokasi kesalahan (*error*) pada kode program sehingga dapat segera diperbaiki oleh pembuat program (*programmer*).

Menurut Rosa dan Shalahuddin (2016: 275) pengujian kotak hitam (*Black-Box Testing*) yaitu menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan.

Pengujian kotak hitam dilakukan dengan membuat kasus uji yang bersifat mencoba semua fungsi dengan memakai perangkat lunak apakah sesuai dengan spesifikasi yang dibutuhkan. Pengujian kotak hitam dilakukan dengan membuat kasus uji yang bersifat mencoba semua fungsi dengan memakai perangkat lunak apakah sesuai dengan spesifikasi yang dibutuhkan. Kasus uji yang dibuat untuk melakukan pengujian kotak hitam harus dibuat dengan kasus benar dan kasus salah, misalkan untuk kasus proses login maka kasus uji yang dibuat adalah:

1. Jika user memasukkan nama pemakai (*username*) dan kata sandi (*password*) yang benar.
2. Jika user memasukkan nama pemakai (*username*) dan kata sandi (*password*) yang salah, misalnya nama pemakai benar tapi kata sandi salah, atau sebaliknya, atau salah keduanya.

2.4 Penelitian Terdahulu

Sebagai bahan pertimbangan dalam penelitian ini, maka penulis mencantumkan beberapa penelitian yang diambil dari beberapa jurnal ilmiah, yaitu:

1. **Dina Andayati** (2012, ISSN: 1979-911X) Sistem Pakar Dalam Bidang Psikologi, Teknologi informasi telah menyebar luas diberbagai bidang dan dengan teknologi informasi akan memudahkan pengguna dalam mencari informasi dan juga mengolah informasi itu sendiri, sehingga mendapatkan sesuatu yang lebih bermanfaat, sesuai dengan kebutuhan pengguna. Sistem pakar dalam penelitian ini didasarkan pada bidang psikologi yang berkaitan

dengan tes psikologi online dengan menggunakan indikator *Myers-Briggs Type Indicator* (MBTI) yang merupakan tes kepribadian yang menggunakan beberapa karakteristik. Aplikasi psikologi yang lain yaitu berhubungan dengan penelusuran minat bakat siswa yaitu dengan metode *Rothwell Miller Inventory Blank* (RMIB). Sistem pakar yang dirancang dapat menganalisa dan menarik kesimpulan, seperti halnya para pakar atau ahli. Sistem pakar tersebut juga dapat membantu menganalisa kepribadian seseorang berdasarkan tes MBTI dan memprediksi pekerjaan yang cocok sesuai minat dan bakat berdasarkan tes RMIB.

2. **Markus Hendrawan SA** (2014, ISSN: 2089-9033) Sistem Pendukung Keputusan Rekrutmen Karyawan di PT Indo Beras Unggul Menggunakan Metode *Analytical Hierarchy Process*, Proses penilaian rekrutmen di PT Indo Beras Unggul dilakukan secara subjektif sehingga menyebabkan hasil akhir perekrutan terhadap calon karyawan dapat mengganggu kandidat yang memiliki kemampuan dan keterampilan tetapi tidak mempunyai nilai subyektifitas yang pada akhirnya tersingkirkan. Selain itu, terdapat adanya perbedaan penilaian berupa angka dan abjad pada tes seleksi sehingga membuat HRD kesulitan dalam menghasilkan nilai hasil seleksi calon karyawan. Sebagai solusi dari permasalahan tersebut adalah dengan membangun sebuah Sistem Pendukung Keputusan Rekrutmen Karyawan. Sistem pendukung keputusan yang dibangun menggunakan metode *Analytical Hierarchy Process* (AHP). AHP merupakan metode pengambilan keputusan yang membantu dalam menentukan prioritas dari beberapa kriteria.

Sistem pendukung keputusan yang dibangun dapat memberi rekomendasi kepada HRD dalam pengambilan keputusan layak atau tidaknya calon karyawan berdasarkan hasil nilai objektif pada penilaian rekrutmen. Selain itu juga dapat memudahkan HRD dalam menghasilkan nilai hasil seleksi sehingga dapat mendapatkan calon karyawan yang tepat.

3. **Mely Amaliyah, Fiftin Noviyanto** (2013, ISSN: 2338-5197) Aplikasi Tes Kepribadian untuk Penempatan Karyawan Menggunakan Metode MBTI (*Myers-Briggs Type Indicator*) Berbasis Web (Studi Kasus: PT Winata Putra Mandiri), Penempatan karyawan merupakan salah satu fungsi terpenting dalam manajemen sumber daya manusia. Penempatan karyawan yang tepat sesuai dengan harapan, bakat dan kepribadian karyawan, akan mampu meningkatkan motivasi, produktivitas dan kepuasan kerja. Salah satu tes penunjang penempatan karyawan adalah tes kepribadian. Dalam penelitian ini, tes kepribadian yang digunakan adalah dengan metode *Myers-Briggs Type Indicator* (MBTI). Aplikasi tes kepribadian tersebut berhasil dibangun dan dapat memberikan hasil tes yang benar sesuai dengan aturan psikologi dan dapat dicetak sebagai laporan serta dapat dilihat oleh karyawan secara cepat dan mudah. Aplikasi tersebut juga dapat mempermudah karyawan dalam melakukan tes.
4. **Putu Veda Andreyana, I Nyoman Piarsa, Putu Wira Buana** (2015, ISSN: 2252-3006) Sistem Pakar Analisis Kepribadian Diri dengan Metode *Certainty Factor*, Kepribadian adalah sifat dan tingkah laku khas seseorang yang membedakan seseorang dengan orang lain. Salah satu metode yang

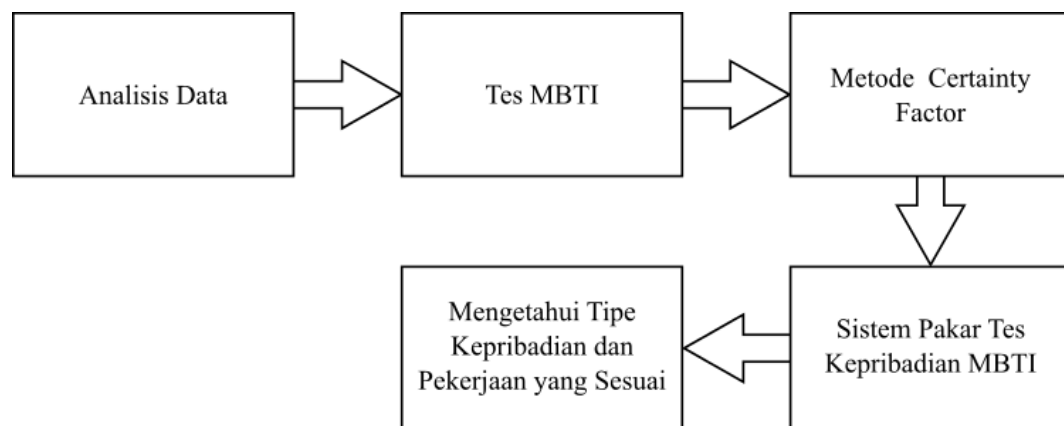
digunakan psikolog adalah *Big Five Personality* untuk mendapatkan jenis-jenis kepribadian. Sistem pakar kepribadian diri dengan metode *Big Five* dapat memudahkan seseorang melakukan tes kepribadian tanpa psikolog dan memudahkan psikolog dalam melakukan tes kepribadian karena tanpa melakukan perhitungan manual. Metode Certainty Factor digunakan dalam sistem pakar analisis kepribadian diri untuk mendapat nilai presentase kepercayaan pakar pada output sistem dengan tingkat akurasi mencapai 74% berdasarkan hasil survey. Sistem pakar analisis kepribadian diri menghasilkan output jenis-jenis kepribadian yang didapat dari jawaban pengguna dengan member pertanyaan-pertanyaan mengenai sifat kepribadian. Output tes kepribadian yang dihasilkan sistem sesuai dengan pengetahuan yang dimiliki oleh pakar psikolog.

5. **Rakhma Indah Fitrianti, Arief Andy Soebroto, dan Barlian Henryranu P** (2012, ISSN: 2089-8185) Sistem Pakar Pada Bidang Teknologi Informasi Untuk Rekomendasi Profesi Pekerjaan Berdasarkan Kepribadian Menggunakan Pendekatan *Personality Factor*, Setiap individu mempunyai hak untuk sukses mencapai karir yang baik. Seseorang memilih karir karena alasan yang salah, mungkin karena keinginan orang tua atau petunjuk guru. Ada juga orang yang memilih karir dengan menerima pekerjaan yang ada meskipun dia tidak nyaman dengan pekerjaan itu dan tidak realistis dalam perencanaan karir. Mengetahui karakter dan kepribadian diri akan mengantarkan seseorang pada kemampuan untuk memilih pekerjaan yang cocok dan menggapai prestasi yang tinggi dalam karir tersebut. Metode

personality factor dengan teknik penalaran *forward chaining* diimplementasikan ke dalam sistem pakar untuk merekomendasikan profesi kerja di bidang teknologi informasi. Sistem pakar yang dihasilkan dapat memberikan hasil 16 kepribadian yang dapat dicocokkan dengan kualifikasi pekerjaan.

2.5 Kerangka Pemikiran

Menurut Sudaryono (2015), kerangka berpikir adalah model konseptual tentang bagaimana teori berhubungan dengan berbagai faktor yang telah diidentifikasi sebagai masalah penting. Berikut ini adalah kerangka pemikiran yang mendasari penelitian ini.



Gambar 2.1 Kerangka pemikiran
Sumber: Data Penelitian (2017)