



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Komputer

Menurut Prawiro (2019) dalam buku *Aplikasi Komputer* karangan Harmayani dkk (2021:1), menjelaskan bahwa Komputer adalah suatu perangkat elektronik yang dapat digunakan untuk mengolah data sesuai dengan prosedur yang telah dirumuskan sebelumnya sehingga menghasilkan informasi bermanfaat bagi penggunaannya.

Menurut Barany dkk (2020:13), menyatakan bahwa komputer adalah sebuah mesin hitung elektronik yang secara cepat menerima informasi masukan digital dan mengolah informasi tersebut menurut seperangkat instruksi yang tersimpan dalam komputer tersebut dan menghasilkan keluaran informasi yang dihasilkan setelah diolah.

Berdasarkan pendapat di atas dapat disimpulkan bahwa komputer adalah suatu perangkat yang dapat mengolah informasi sesuai dengan prosedur yang tersimpan dan menghasilkan informasi bermanfaat yang telah diolah.

2.1.2 Pengertian Perangkat Lunak (*Software*)

Menurut Seliwati (2022:22), Pengertian *Software* adalah program komputer yang menjadi jembatan antara pengguna dengan perangkat keras (*hardware*).

Menurut Iskandar (2018:68), *Software* adalah bagian sistem komputer yang tidak memiliki wujud, *software* juga bisa memiliki pengertian sebagai data yang berformat digital dan disimpan secara digital yang hanya bisa dibaca oleh komputer.



Berdasarkan pendapat tersebut dapat disimpulkan bahwa *software* (perangkat lunak) adalah program komputer yang menjalankan perintah tetapi tidak terlihat secara fisik yang hanya bisa dibaca oleh komputer dan tersimpan secara digital.

2.1.3 Pengertian Web

Menurut Hidayat (2021:14), *Web* dapat diartikan sebagai kumpulan halaman yang berisi informasi yang dapat diambil melalui media internet berupa tulisan, gambar, suara, video yang memiliki alamat pada browser.

Menurut Gregorius (2000) dalam buku *E-Commerce: Suatu Pengantar Bisnis Digital* karangan Harmayani dkk (2020:42), menjelaskan bahwa *Website* adalah kumpulan halaman *web* yang saling terhubung dan file-filenya saling terkait. *Web* terdiri dari *page* atau halaman, dan kumpulan halaman yang dinamakan *homepage*.

Berdasarkan pendapat di atas dapat disimpulkan bahwa *website* adalah halaman informasi yang bisa diakses oleh seluruh dunia dan seluruh filenya saling terkait.

2.1.4 Pengertian Basis Data

Menurut Suryansah, dkk (2020:97), Basis Data adalah kumpulan informasi yang disimpan di dalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut.

“Basis data (*database*) merupakan kumpulan data yang saling berhubungan (punya relasi).” (Alda, 2020:29)



Dari pendapat tersebut dapat disimpulkan bahwa Basis Data adalah sekumpulan data terhubung yang disimpan di dalam komputer untuk mendapatkan suatu informasi.

2.1.5 Pengertian Sistem Informasi

Menurut (Gaol, Jimmy, 2008) dalam buku *Sistem Informasi Manajemen* karangan Sudirman, dkk (2020:2), yang menyatakan bahwa sistem informasi adalah sebuah sistem yang berguna untuk kepentingan organisasi dalam bentuk pengolahan transaksi dan pengolahan informasi untuk fungsi manajemen dalam sebagai dasar pengambilan keputusan.

Menurut Supartha, dkk (2022:2), Sistem informasi adalah suatu sistem yang mengkombinasikan antara aktivitas manusia dan pemanfaatan teknologi dengan tujuan untuk mendukung manajemen dan kegiatan operasional, berupa sistem yang menyediakan semua informasi yang bermanfaat untuk seluruh tingkatan dalam organisasi, sistem informasi yang bagus harus bisa menyajikan data yang valid dan berkualitas yang bisa diakses kapan saja.

Dari pendapat di atas dapat disimpulkan bahwa sistem informasi merupakan kombinasi antara manusia dengan teknologi yang berguna untuk kepentingan organisasi dan menghasilkan data yang valid.

2.2 Teori Khusus

2.2.1 Pengertian *Unified Modeling Language* (UML)

Menurut Mulyani (2017:35), mengatakan bahwa UML adalah sebuah teknik pengembangan sistem yang menggunakan bahasa grafis sebagai alat untuk pendokumentasian dan melakukan spesifikasi pada sistem.

UML adalah bahasa untuk menspesifikasi, memvisualisasi, membangun dan mendokumentasikan artifacts (bagian dari informasi yang digunakan untuk dihasilkan oleh proses pembuatan perangkat lunak, artifact tersebut dapat berupa model, deskripsi atau perangkat lunak) dari sistem perangkat lunak, seperti pada



pemodelan bisnis dan sistem non perangkat lunak lainnya. (Destriana, 2021:1-2)

Berdasarkan pendapat tersebut dapat disimpulkan bahwa *Unified Modeling Language* (UML) adalah bahasa pemodelan untuk proses pembuatan perangkat lunak.

2.2.2 Pengertian Use Case Diagram

Menurut Tohari (2014) dalam buku *Analisa dan Perancangan Sistem Informasi: Bintang Pustaka* karangan Muharni (2021:54) menjelaskan bahwa *Use Case* adalah rangkaian atau urutan sekelompok yang saling terkait dan membentuk sistem secara teratur yang dilakukan atau diawasi oleh sebuah actor.

Menurut Nugroho, dkk (2021:44), *Use Case Diagram* merupakan diagram yang harus dibuat pertama kali saat pemodelan perangkat lunak berorientasi objek dilakukan.

Berdasarkan pendapat tersebut dapat disimpulkan bahwa *Use Case Diagram* adalah diagram yang pertama kali dibuat dan membentuk sistem secara teratur lalu diawasi oleh sebuah actor.



Tabel 2.1 Simbol Use Case Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i>
2		<i>Dependency</i>	Hubungan di mana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan memengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>).
3		<i>Generalization</i>	Hubungan di mana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
4		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara eksplisit.
5		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas



Lanjutan tabel 2.1 Simbol Use Case Diagram

8		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu <i>actor</i>
9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah elemen-elemennya (sinergi).
10		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

(Sumber : Negara, 2021:79-80)

2.2.3 Pengertian *Class Diagram*

Menurut Negara, dkk (2021:80), *Class Diagram* merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem, juga memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem.

Menurut Putra, dkk (2020:13), *Class diagram* adalah suatu diagram yang dapat menggambarkan variabel atau atribut suatu kelas pada sistem beserta metode yang dapat dilakukan oleh kelas tersebut.

Berdasarkan pendapat tersebut dapat disimpulkan bahwa *Class Diagram* adalah sebuah diagram yang memiliki hubungan antar kelas yang menentukan perilaku sistem.



Tabel 2.2 Simbol-Simbol *Class Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
2		<i>Nary Association</i>	Upaya yang digunakan untuk menghindari asosiasi yang lebih dari 2 objek.
3		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor
5		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6		<i>Dependency</i>	Hubungan di mana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempegaruhi elemen yang bergantung padanya (elemen yang tidak mandiri)
7		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

(Sumber : Sa'ad (2020:50-51))



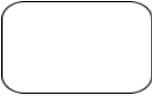
2.2.4 Pengertian Activity Diagram

Menurut Pressman (2010), dalam buku *Otodidak Web Programming: Membuat Website Edutainment* karangan Saad (2020:51), sebuah *activity diagram* menggambarkan perilaku dinamis dari sistem atau bagian dari sistem melalui aliran kontrol antara tindakan yang sistem lakukan.

Menurut Dennis (2009, 158) dalam buku *Algoritma Klasifikasi Bibit Terbaik untuk Tanaman Keladi Tikus* karangan Hesnananda (2021:26), *activity diagram* merupakan model analisis yang digunakan atau menggambarkan sebuah proses aktivitas.

Berdasarkan pendapat di atas dapat disimpulkan bahwa *Activity Diagram* adalah aliran dari sistem yang menggambarkan sebuah proses aktivitas.

Tabel 2.3 Simbol Activity Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Activity</i>	Memperlihatkan bagaimana kelas antarmuka saling berinteraksi satu sama lain
2		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
4		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
5		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

(Sumber : Sa'ad (2020:51-52))



2.2.5 Pengertian *Sequence Diagram*

Menurut Muhammad (2011), dalam buku *Otodidak Web Programming: Membuat Website Edutainment* karangan Saad (2020:52), *Sequence diagram* adalah suatu diagram yang menggambarkan interaksi antar objek dan mengindikasikan komunikasi di antara objek-objek tersebut.

Herlinah dan Musliadi (2019:54) menjelaskan bahwa *Sequence Diagram* merupakan salah satu dari beberapa diagram yang ada pada UML. *Sequence diagram* merupakan diagram yang menggambarkan atau mendeskripsikan bagaimana entitas dalam sistem berinteraksi, kolaborasi dinamis antara sejumlah objek yang digunakan sebagai wujud rangkaian pesan yang dikirim antara object juga interaksi antara objek.

Berdasarkan pendapat di atas dapat disimpulkan bahwa *Sequence Diagram* adalah diagram UML yang mendeskripsikan bagaimana entitas dalam sistem berinteraksi di antara objek-objek tersebut.

Tabel 2.4 Simbol *Sequence Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1.		Aktor	Aktor merupakan pengguna yang melakukan interaksi dengan system.
2.		Lifeline	Objek entity, antarmuka yang saling berinteraksi.
3.		Message	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi



Lanjutan Tabel 2.4 Simbol *Sequence Diagram*

4.		Message	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi
----	---	---------	--

2.3 Teori Judul

2.3.1 *Preference Ranking Organization Method for Enrichment Evaluation (PROMETHEE)*

Sari (2018:113), *Promethee* adalah suatu penentuan urutan (Prioritas) dalam analisis multikriteria. Masalah pokoknya adalah kesederhanaan, kejelasan dan kestabilan. Dugaan dari dominasi kriteria yang digunakan dalam *Promethee* adalah penggunaan nilai dalam hubungan *Outranking*. *Promethee* termasuk metode peringkat yang cukup sederhana dalam konsep dan aplikasi dibandingkan dengan metode lain untuk analisis multikriteria.

Nofriansyah dan Defit (2017:106-107) menjelaskan bahwa prinsip yang digunakan adalah penetapan prioritas *alternative* yang telah ditetapkan berdasarkan pertimbangan $(\forall i | f_i(\cdot) \rightarrow R [realworld])$, dengan kaidah dasar: $\text{Max} \{f_1(X), f_2(X), f_3(X), \dots, f_i(X), \dots, f_k(X) | X \in R\}$,

Dimana K adalah sejumlah kumpulan *alternative*, dan f_i ($i=1,2,3,\dots,K$) merupakan nilai/ukuran *relative criteria* masing-masing *alternative*. Dalam aplikasinya sejumlah kriteria telah ditetapkan untuk menjelaskan K yang merupakan penilaian dari R (*real*).



Promethee termasuk dalam keluarga metode *Outranking* yang dikembangkan oleh B. Roy dan meliputi dua fase :

- Membangun hubungan *Outranking* dari K.
- Eksploitasi dari hubungan ini memberikan jawaban optimasi kriteria dalam paradigma permasalahan multikriteria.

Dalam fase pertama, nilai hubungan *outranking* berdasarkan pertimbangan dominasi masing-masing kriteria indeks preferensi ditentukan dan nilai *outranking* secara grafis disajikan berdasarkan preferensi dari pembuat keputusan. Data dasar untuk evaluasi dengan metode *Promethee* disajikan sebagai berikut (Daihani dan Dadan, 2001).

Tabel 2.5 Data Dasar Analisis *Promethee*

Kriteria	Min Max	Alternatif						LF	EF	NF
		A1	A2	A3	A4	A5	A6			
K1	Min	K1(A1)	K1(A2)	K1(A3)	K1(A4)	K1(A5)	K1(A6)			
K2	Max	K2(A1)	K2(A2)	K2(A3)	K2(A4)	K2(A5)	K2(A6)			
K3	Max	K3(A1)	K3(A2)	K3(A3)	K3(A4)	K3(A5)	K3(A6)			
K4	Max	K4(A1)	K4(A2)	K4(A3)	K4(A4)	K4(A5)	K4(A6)			
K5	Max	K5(A1)	K5(A2)	K5(A3)	K5(A4)	K5(A5)	K5(A6)			
K6	Max	K6(A1)	K6(A2)	K6(A3)	K6(A4)	K6(A5)	K6(A6)			

Sumber (Nofriansyah dan Defit (2017:106-107))

Berikut ini merupakan penjelasan dari istilah atau singkatan yang digunakan :

1. K1(A1) : Elemen matrik K1 baris ke 1 dan kolom ke 1
2. K1(A2) : Elemen matrik K1 baris ke 1 dan kolom ke 2
3. Kmn : Elemen matrik K baris ke m dan kolom ke n



2.3.1.1 Dominasi Kriteria

Nilai f merupakan nilai nyata dari suatu kriteria dan tujuan berupa prosedur optimasi : $f: K \rightarrow R$. Untuk setiap alternatif $a \in K$, $f(a)$ merupakan evaluasi dari alternatif tersebut untuk suatu kriteria. Pada alternatif dibandingkan, $a_1, a_2 \in K$, harus dapat ditentukan perbandingan preferensinya.

Menurut Brans penyampaian intensitas (P) dari preferensi alternatif a terhadap alternatif a sedemikian rupa sehingga :

- a. $P(a_1, a_2) = 0$, berarti tidak ada beda antara a_1 , dan a_2 , atau tidak ada preferensi dari a_1 , lebih baik dari a_2 .
- b. $P(a_1, a_2) \sim 0$, berarti lemah preferensi dari a_1 lebih baik dari a_2 .
- c. $P(a_1, a_2) \sim 1$, berarti mutlak, preferensi dari a_1 lebih baik dari a_2 .
- d. $P(a_1, a_2) = 1$, berarti mutlak, preferensi dari a_1 lebih baik dari a_2 .

2.3.1.2 Fungsi Preferensi

Dalam rangka memfasilitasi pemilihan fungsi preferensi tertentu, Brans dan Vinckdle, mengusulkan enam tipe preferensi yaitu :

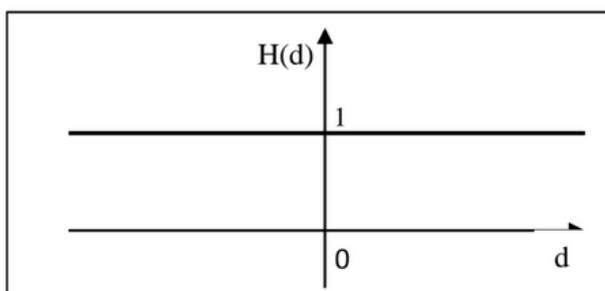
1. Kriteria Biasa (Usual Criterion)

$$H(d) \begin{cases} 0 & \text{jika } d = 0 \\ 1 & \text{jika } d \neq 0 \end{cases}$$

dimana $d =$ selisih nilai kriteria $\{d = f(a) - f(b)\}$



Pada kasus ini tidak ada beda (sama penting) antara a dan b jika hanya jika $f(a) = f(b)$. Apabila nilai kriteria pada masing-masing alternatif memiliki nilai berbeda, pembuat keputusan membuat preferensi mutlak untuk alternatif yang memiliki nilai lebih baik.



Gambar 2.1 Usual Criterion

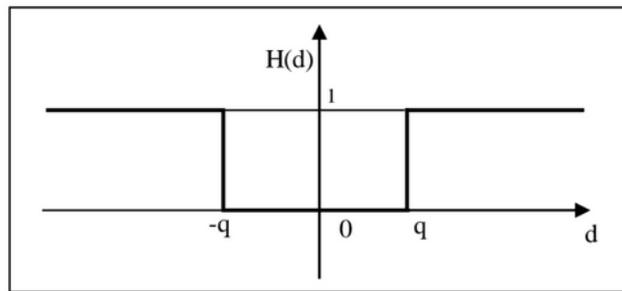
2. Kriteria Quasi (*Quasi Criterion*)

$$H(d) \begin{cases} 0 & \text{jika } -q \leq d \leq q \\ 1 & \text{jika } d < -q \text{ atau } d > q \end{cases}$$

Keterangan :

1. $H(d)$: fungsi selisih kriteria antar alternatif
2. d : selisih nilai kriteria $\{d = f(a) - f(b)\}$
3. Parameter (q) : harus merupakan nilai yang tetap

Alternatif memiliki preferensi yang sama penting jika selisih dari masing-masing alternatif untuk kriteria tertentu tidak melebihi nilai q . Apabila selisih masing-masing alternatif melebihi nilai q , maka bentuk preferensi mutlak

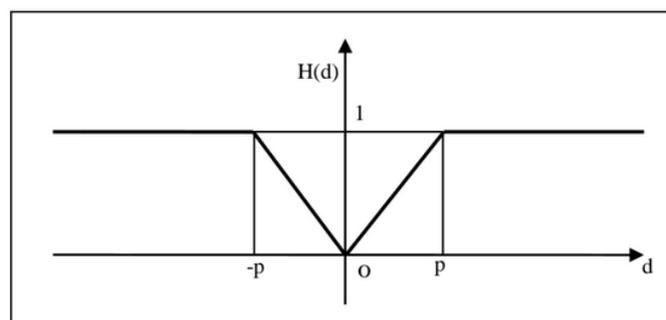


Gambar 2.2 Quasi Criterion

3. Kriteria Preferensi Linier

$$H(d) = \begin{cases} 0 & \text{jika } -p \leq d \leq p \\ 1 & \text{jika } d < -p \text{ atau } d > p \end{cases}$$

Kriteria preferensi linier dapat menjelaskan bahwa selama nilai selisih memiliki nilai yang lebih rendah dari p, preferensi dari pembuat keputusan meningkat secara linier dengan nilai d. Jika nilai d lebih besar dibandingkan dengan nilai p, maka terjadi preferensi mutlak. Pada saat pembuat keputusan mengidentifikasi beberapa kriteria untuk tipe ini, dia harus menentukan nilai dari kecenderungan atas (nilai p). Dalam hal ini nilai d di atas p telah dipertimbangkan akan memberikan preferensi mutlak dari satu alternatif.



Gambar 2.3 Preferensi Linear



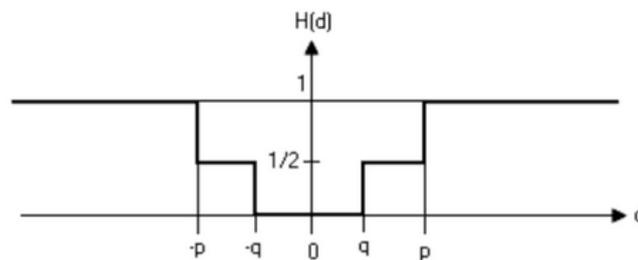
4. Kriteria Level

Dalam kasus ini, kecenderungan tidak berbeda dengan q dan kecenderungan preferensi ditentukan secara simultan. Jika d berada di antara nilai p dan q , hal ini berarti situasi preferensi yang lemah ($H(d) = 0,5$). Fungsi ini disajikan pada :

$$H(d) = \begin{cases} 0 & \text{jika } d \leq q \\ 0,5 & \text{jika } q < d \leq p \\ 1 & \text{jika } d > p \end{cases}$$

Keterangan :

1. $H(d)$: fungsi selisih antara alternatif
2. P : nilai kecenderungan atas
3. Parameter (q) : harus merupakan nilai yang tetap



Gambar 2.4 Kriteria Level

5. Kriteria Linear dan Area yang Tidak Berbeda

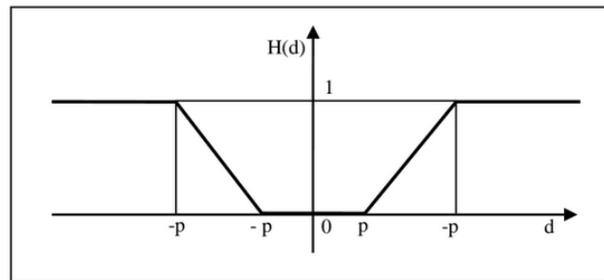
Pada kasus ini, pengambil keputusan mempertimbangkan peningkatan preferensi secara linear dari tidak berbeda hingga preferensi mutlak dalam area dua kecenderungan q dan p .

$$H(d) = \begin{cases} 0 & \text{jika } d \leq q \\ (d - q) / (p - q) & \text{jika } q < d \leq p \\ 1 & \text{jika } d > p \end{cases}$$



1. $H(d)$: fungsi selisih antara alternatif
2. d : selisih nilai kriteria $\{d = f(a) - f(b)\}$
3. parameter (p) : nilai kecenderungan atas
4. parameter (q) : harus merupakan nilai yang tetap

Dua parameter p dan q telah ditentukan nilainya. Fungsi $H(d)$ adalah hasil perbandingan antara alternatif, seperti pada gambar 2.5

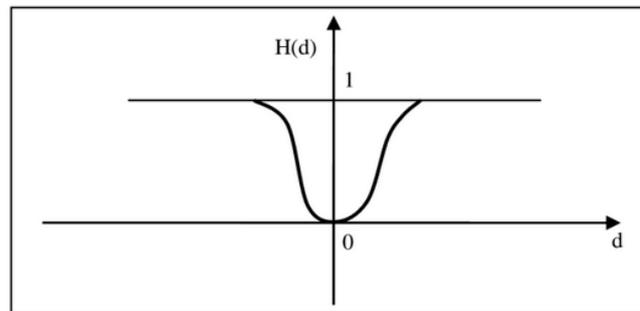


Gambar 2.5 Kriteria Linear dan Area yang Tidak Berbeda

6. Kriteria Gaussian

Fungsi ini bersyarat apabila telah ditentukan nilai σ , dimana dapat dibuat berdasarkan distribusi normal dalam statistik.

$$H(d) = 1 - \exp \{-d^2/2\sigma^2\}$$



Gambar 2.6 Kriteria Gaussian

2.3.1.3 Indeks Preferensi Multikriteria

Tujuan pembuat keputusan adalah menteapkan fungsi preferensi P_i , dan π_i untuk semua kriteria f_i ($i = 1, 2, 3, \dots, K$) dari masalah optimasi kriteria majemuk. Bobot (weight) π_i merupakan ukuran *relative* untuk kepentingan kriteria f_i , jika semua kriteria memiliki kepentingan yang sama dalam pengambilan keputusan maka semua nilai bobot adalah sama.

Indeks preferensi multikriteria ditentukan berdasarkan rata-rata bobot dari fungsi preferensi P_i .

$$\Phi(\mathbf{a}, \mathbf{b}) = \sum_{i=1}^n \pi_i P_i(a, b); \forall a, b \in A$$

$\Phi(\mathbf{a}, \mathbf{b})$ merupakan intensitas preferensi pembuat keputusan yang menyatakan bahwa alternatif a lebih baik dari alternatif b dengan pertimbangan secara simultan dari seluruh kriteria. Hal ini dapat disajikan dengan nilai antara nilai 0 dan 1, dengan ketentuan sebagai berikut :

1. $\Phi(\mathbf{a}, \mathbf{b}) = 0$ menunjukkan preferensi yang lemah untuk alternatif lebih dari alternatif berdasarkan semua kriteria.



2. $\Phi(\mathbf{a}, \mathbf{b}) = 1$ menunjukkan preferensi yang kuat untuk alternatif lebih dari alternatif berdasarkan semua kriteria

2.3.1.4 Tahapan Prosedur *Promethee*

Adapun tahapan prosedur untuk pelaksanaan *Promethee* adalah sebagai berikut: (Ignatius, 2012)

1. Penentuan deviasi berdasarkan perbandingan berpasangan $d_j(\mathbf{a}, \mathbf{b}) = g(\mathbf{a}) - g(\mathbf{b})$ $j = 1, 2, \dots, k$

Dimana $d_j(\mathbf{a}, \mathbf{b})$ menunjukkan perbedaan antara evaluasi dari \mathbf{a} dan \mathbf{b} pada setiap kriteria, dan k menunjukkan kriteria berhingga.

2. Penerapan fungsi preferensi

$$P_j(\mathbf{a}, \mathbf{b}) = F_j(d_j(\mathbf{a}, \mathbf{b})) \quad j = 1, 2, \dots, k$$

Dimana $P_j(\mathbf{a}, \mathbf{b})$ sebagai fungsi $d_j(\mathbf{a}, \mathbf{b})$ menunjukkan preferensi alternatif \mathbf{a} yang berkaitan dengan alternatif \mathbf{b} pada setiap kriteria.

3. Perhitungan indeks preferensi global

$$\Phi(\mathbf{a}, \mathbf{b}) = \sum_{j=1}^n P_j(\mathbf{a}, \mathbf{b}) W_j \quad \forall \mathbf{a}, \mathbf{b} \in A$$

Dimana $\Phi(\mathbf{a}, \mathbf{b})$ dengan \mathbf{a} lebih besar dari \mathbf{b} (antara nol hingga satu) didefinisikan sebagai jumlah bobot $p(\mathbf{a}, \mathbf{b})$ pada setiap kriteria, dan w_j adalah bobot yang berhubungan dengan Kriteria ke- j .

4. Perhitungan aliran perangkingan dan peringkat parsial

$$\varphi^+(\mathbf{a}) = \frac{1}{n-1} \sum_{x \in A} \phi(\mathbf{a}, \mathbf{b})$$

$$\varphi^-(\mathbf{a}) = \frac{1}{n-1} \sum_{x \in A} \phi(\mathbf{a}, \mathbf{b})$$

dimana $\varphi^+(\mathbf{a})$ dan $\varphi^-(\mathbf{a})$ menunjukkan *leaving flow* dan *entering flow*



pada setiap alternatif.

5. Perhitungan aliran perangkungan bersih dan peringkat lengkap

$$\varphi(a) = \varphi^+(a) - \varphi^-(a)$$

dimana $\varphi(a)$ adalah *net flow*, digunakan untuk menghasilkan keputusan akhir penentuan urutan dalam menyelesaikan masalah sehingga menghasilkan urutan lengkap.

Dalam metode *Promethee* ada dua macam perangkungan parsial yang didasarkan pada hasil perhitungan yakni sebagai berikut :

1. Perangkungan parsial yang didasarkan pada nilai *Entering Flow* dan *Leaving Flow*.
2. Perangkungan lengkap atau komplit yang didasarkan pada nilai *Net Flow*.

2.3.2 Ketenagakerjaan

Sembiring (2016:2-28), Perjanjian kerja merupakan dasar bagi pekerja dan juga dasar bagi perusahaan untuk mengubah status dari pekerja tersebut, yakni dari pekerja kontrak menjadi pekerja tetap. Mengingat pentingnya perjanjian kerja, sudah sepantasnya masing-masing pihak memerhatikan tiap-tiap aturan yang dicantumkan di dalam perjanjian kerja.

2.3.2.1 Perjanjian Kerja Waktu Tertentu

Perjanjian kerja waktu tertentu (PKWT) adalah perjanjian kerja yang didasarkan pada suatu pekerjaan yang penyelesaian dapat diperkirakan, sedangkan perjanjian kerja waktu tidak tentu (PPKWTT) adalah perjanjian kerja yang pekerjaannya tidak dapat diperkirakan penyelesaiannya.



Pada UU No. 13 Tahun 2003 tentang Ketenagakerjaan, disyaratkan adanya masa percobaan bagi pekerja. Masa percobaan tersebut hanya berlaku terhadap perjanjian kerja waktu tidak tentu (PKWTT), sedangkan pada perjanjian kerja waktu tertentu (PKWT), tidak diperkenankan adanya masa percobaan. Selain itu, pada perjanjian kerja waktu tertentu, pekerjaan yang dilakukan haruslah pekerjaan yang bersifat tidak tetap. Penjelasan Pasal 59 UU No. 13 Tahun 2003 tentang Ketenagakerjaan mengenai pekerjaan yang bersifat tetap sebagai berikut.

Yang dimaksud dengan pekerjaan yang bersifat tetap dalam ayat ini adalah pekerjaan yang sifatnya terus menerus, tidak terputus-putus, tidak dibatasi waktu, dan merupakan bagian dari suatu proses produksi dalam satu perusahaan atau pekerjaan yang bukan musiman.

Adapun jenis-jenis dari Perjanjian Kerja Waktu Tertentu (PKWT) sebagai berikut :

- Pekerjaan yang sekali selesai atau sementara sifatnya yang penyelesaiannya paling lama 3 tahun.
- Bersifat musiman, yakni bergantung pada musim/cuaca dan untuk memenuhi pesanan/target tertentu.
- Berhubungan dengan produk baru, kegiatan baru, atau produk tambahan yang masih dalam percobaan atau peninjauan.
- Pekerja harian/lepas untuk pekerjaan yang berubah-ubah dalam hal waktu dan volume dan waktu kerja kurang dari 21 hari dalam satu bulan.



2.3.2.2 Perpanjangan Perjanjian Kerja Waktu Tidak Tertentu (PKWTT)

Dasar dari pekerjaan yang berdasarkan perjanjian kerja waktu tertentu adalah adanya suatu pekerjaan yang pasti akan selesai dalam jangka waktu tertentu. Meskipun demikian, hal tersebut dapat saja meleset, sehingga jika tidak dilakukan suatu upaya, tentunya pekerjaan tersebut tidak dapat diselesaikan dan jika harus merekrut tenaga kerja yang baru tentunya akan membutuhkan waktu.

Pasal 59 ayat (4) menentukan bahwa perjanjian kerja waktu tertentu hanya dapat diperpanjang untuk jangka waktu paling lama 2 (dua) tahun dan dapat diperpanjang untuk jangka waktu paling lama 1 (satu) tahun, sehingga total keseluruhan perjanjian kerja waktu tertentu adalah 3 (tiga) tahun.

2.3.2.3 Perjanjian Kerja Waktu Tidak Tertentu (PKWTT)

Perjanjian kerja waktu tidak tertentu (PKWTT) adalah perjanjian kerja yang didasarkan pada pekerjaan yang bersifat pekerjaan khusus, misalnya pekerjaan untuk membangun jembatan dan pekerjaan untuk mengoperasikan jembatan.

Pada perjanjian kerja waktu tertentu, untuk dapat diangkat menjadi karyawan permanen, disyaratkan adanya masa percobaan selama 3 (tiga) bulan dan jika kinerja pekerja yang bersangkutan memenuhi persyaratan, status dari pekerja yang bersangkutan berubah dari pekerja kontrak menjadi pekerja permanen.

Dalam masa 3 (tiga) bulan, kemampuan kerja dari pekerja yang bersangkutan benar-benar dinilai, dari tingkat kedisiplinan, misalnya jam masuk dan pulang kantor, hingga terobosan baru apa yang dilakukan oleh pekerja yang bersangkutan, bagaimana pekerja yang bersangkutan menyelesaikan pekerjaannya, berapa lama pekerja yang bersangkutan menyelesaikan pekerjaannya, bagaimana hubungan pekerja yang bersangkutan dengan pekerja lainnya. Dengan demikian, dapat dikatakan bahwa dalam masa 3 (tiga) bulan



tersebut, pekerja yang bersangkutan diperhatikan dari segala macam aspek yang mana hal ini tentunya bertujuan untuk dapat menilai apakah pekerja yang bersangkutan sesuai dengan kebutuhan perusahaan atau tidak, apakah pekerja yang bersangkutan dapat mengikuti budaya kerja perusahaan atau tidak.

Jika pekerja berubah statusnya dari karyawan kontrak menjadi karyawan tetap, perusahaan akan menerbitkan surat pengangkatan sebagai karyawan tetap. Berdasarkan surat tersebut, secara hukum status dari karyawan tersebut berubah menjadi karyawan tetap.

Dengan berubahnya status dari pekerja tersebut, jika terjadi pemutusan hubungan kerja, pemutusan hubungan kerja tersebut tidak mengacu kepada perjanjian kerja, tetapi mengacu pada UU No. 13 Tahun 2003 tentang Ketenagakerjaan. Pada undang-undang tersebut, telah ditentukan hal-hal apa saja yang harus dipenuhi agar dapat dilakukan pemutusan hubungan kerja serta hal-hal apa saja yang menjadi hak-hak dari pekerja yang mengalami pemutusan hubungan kerja.

2.3.2.4 Hal-Hal yang Perlu Diperhatikan dalam Perjanjian Kerja

Sebelum perjanjian kerja ditandatangani, harus diperhatikan dengan saksama hal-hal lainnya, seperti pemutusan hubungan kerja. Hal ini penting sebab hal inilah yang mendasari jika terjadi pemutusan perjanjian. Mengenai pemutusan hubungan kerja, setidaknya dapat terjadi disebabkan adanya pemutusan hubungan kerja oleh perusahaan yang dapat didasarkan oleh berbagai macam sebab, salah satunya karena adanya efisiensi oleh perusahaan, pemutusan hubungan kerja oleh pegawai, atau pengunduran diri yang mana dapat disebabkan oleh ketidakcocokan karyawan yang bersangkutan dalam bekerja di perusahaan tersebut.



2.3.2.5 Pekerja Kontrak

Pada suatu perusahaan, status karyawan/pekerjanya dibagi menjadi 2 (dua), yakni karyawan tetap dan karyawan kontrak. Karyawan yang tetap bukanlah karyawan yang pada awal mula bekerja langsung mendapatkan status tetap, tetapi harus melewati tahapan sebagai pekerja kontrak terlebih dahulu, meskipun tidak tertutup kemungkinan bahwa karyawan tersebut langsung dapat menjadi karyawan tetap.

Misalnya, karyawan tersebut adalah seseorang yang diminati oleh perusahaan dan ditawarkan untuk bekerja. Karyawan tersebut mengajukan persyaratan bahwa dia bersedia untuk bekerja di perusahaan tersebut dengan catatan langsung menjadi karyawan tetap. Sebagaimana yang sudah dijelaskan terlebih dahulu, bahwa perjanjian kerja dibagi menjadi 2 (dua), yakni perjanjian kerja waktu tertentu (PKWT) dan perjanjian kerja waktu tak tentu (PKWTT). Pada perjanjian kerja waktu tertentu, hubungan kerja berakhir jika jangka waktu yang disepakati telah tercapai, sedangkan pada perjanjian kerja waktu tak tentu, berakhirnya hubungan kerja yang disebabkan adanya pemutusan hubungan kerja, jadi tidak didasarkan oleh waktu.

2.3.2.6 Pengangkatan dan Pemberhentian Karyawan

Ardiputra, dkk (2022:5) menjelaskan bahwa pada proses ini kegiatan pengisian formasi yaitu merekrut karyawan, screening, serta seleksi kepada pelamar juga penempatan formasi. Fungsi manajemen sumber daya manusia semakin rumit karena berbagai peraturan memberi peluang yang sama untuk seluruh lapisan masyarakat dalam mendapatkan pekerjaan serta semakin kompleksnya formasi jabatan yang akan diisi.



Proses pengangkatan dan pemberhentian (staffing sesuai dengan kebutuhan perusahaan) ini meliputi :

- *Recruitment* calon pelamar pekerjaan.
- Seleksi calon pekerja atau karyawan yang sesuai dengan persyaratan yang dibutuhkan perusahaan.
- Mengadakan proses pemutusan hubungan kerja dengan baik kepada karyawan.

Proses *recruitment* dan evaluasi pekerjaan di antaranya ialah blanko lamaran, *interview*, tes psikologi, serta pengecekan referensi. Ketika sudah berakhir masa jabatan atau masa kerjanya, maka perusahaan memiliki tanggung jawab harus melakukan pemutusan hubungan kerja dengan karyawan dengan sebaik-baiknya.

2.4 Teori Program

2.4.1 Pengertian HTML

Menurut Setiawan (2017:16), HTML atau *Hyper Text Markup Language* merupakan sebuah bahasa pemrograman terstruktur yang dikembangkan untuk membuat halaman *website* yang dapat diakses atau ditampilkan menggunakan *Web Browser*.

Menurut Sitepu (2018:1), HTML adalah singkatan dari *Hypertext Markup Language* yang merupakan sebuah bahasa markup pada internet khususnya web berupa kode dan simbol. Kode dan simbol tersebut nantinya akan ditempatkan ke dalam sebuah file dengan tujuan untuk dimunculkan pada sebuah *website*.



Dari pendapat tersebut dapat disimpulkan bahwa HTML adalah bahasa pemrograman terstruktur yang bertujuan untuk ditampilkan pada sebuah *website* yang dapat diakses dan digunakan untuk mendesain sebuah *website*.

2.4.2 Pengertian PHP

PHP merupakan bahasa pemrograman yang digunakan untuk membuat *website* dinamis dan interaktif. Dinamis artinya, *website* tersebut bisa berubah-ubah tampilan dan kontennya sesuai kondisi tertentu. (Jubilee Enterprise, 2018:1)

Putratama (2018:3), PHP (PHP: *Hypertext Preprocessor*) adalah suatu bahasa pemrograman yang digunakan untuk menerjemahkan baris kode program menjadi kode mesin yang dapat dimengerti oleh komputer yang bersifat *server-side* yang dapat ditambahkan ke dalam HTML.

Dari pendapat tersebut dapat disimpulkan bahwa PHP merupakan bahasa pemrograman yang digunakan untuk membuat *web* dinamis dan interaktif. PHP juga dapat digunakan untuk menerjemahkan baris kode program menjadi kode mesin yang dimengerti oleh komputer.

2.4.3 Pengertian XAMPP

Andhara, dkk (2022:14), Pengertian XAMPP adalah perangkat lunak (*free software*) bebas, yang mendukung banyak sistem operasi, dan merupakan kompilasi dari beberapa program.

Habibi, dkk (2020:5) menjelaskan bahwa XAMPP adalah perangkat yang menggabungkan tiga aplikasi ke dalam satu paket yaitu Apache, MySQL, dan PHP my admin dengan xampp pekerjaan anda sangat dimudahkan karena dapat menginstalasi dan mengkonfigurasi ketiga aplikasi tersebut dengan sekaligus dan otomatis. XAMPP merupakan salah satu paket instalasi apache, php dan MySQL instan yang dapat kita gunakan untuk membantu proses instalasi ketiga tersebut selain paket instalasi instan xampp untuk berpindah versi juga memberikan fasilitasi pilihan penggunaan php.



Dari pendapat tersebut dapat disimpulkan bahwa XAMPP adalah perangkat lunak yang mendukung banyak sistem operasi dan menggabungkan apache, php, MySQL ke dalam satu paket. XAMPP juga dapat menginstallasi ketiga aplikasi tersebut dengan otomatis.

2.4.4 Pengertian MySQL

Menurut Indrawan (2021:3), MySQL adalah sebuah program *database server* yang mampu menerima dan mengirimkan datanya sangat cepat, multi user serta menggunakan perintah dasar SQL (*Structured Query Language*).

Menurut Enterprise (2018:2), MySQL merupakan *server* yang melayani *database*. Untuk membuat dan mengolah *database*, kita dapat mempelajari pemrograman khusus yang disebut *query* (perintah) SQL.

Database sendiri dibutuhkan jika kita ingin menginput data dari *user* menggunakan *form* HTML untuk kemudian diolah PHP agar bisa disimpan ke dalam *database* MySQL.

Dari pendapat tersebut dapat disimpulkan bahwa MySQL adalah program *database server* yang mengolah database dan dapat bekerja dengan sangat cepat menggunakan pemrograman khusus yaitu perintah dasar SQL.

2.4.5 Pengertian phpMyAdmin

Menurut Yudhanto (2018:12), PhpMyAdmin adalah aplikasi berbasis *web* yang digunakan untuk melakukan pengelolaan *database* MySQL dan atau tool yang paling populer untuk mengelola *database* MySQL.

Menurut Yudhanto dan Prasetyo (2019:19), PhpMyAdmin adalah aplikasi *web* untuk mengelola *database* MySQL dan *database* MariaDB dengan mudah melalui antarmuka (*interface*) grafis. Aplikasi *web* ini ditulis menggunakan bahasa pemrograman PHP. Sebagaimana aplikasi-aplikasi lain untuk lingkungan *web* (aplikasi yang dibuka atau dijalankan menggunakan *browser*), phpMyAdmin juga



mengandung unsur HTML/XHTML, CSS, dan juga kode JavaScript.

Dari pendapat tersebut dapat disimpulkan bahwa phpMyAdmin adalah aplikasi web yang dapat mengelola database MySQL dengan mudah melalui *interface* grafis.

2.5 Metode Pengembangan Sistem

2.5.1 Metode *eXtreme Programming*

Menurut Subakti, dkk (2022:29-30), *Extreme Programming* adalah metodologi proses pengembangan perangkat lunak dalam RPL. Metode ini adalah merupakan sebuah metode pengembangan perangkat lunak yang sering dimanfaatkan dalam proses pengembangan perangkat lunak. Tujuan utama dari pendekatan ini adalah untuk mengurangi biaya dan menyederhanakan fase yang berbeda dari proses pengembangan perangkat lunak.

Menurut Suryantara (2017:24), XP memiliki kesan kumpulan ide lama yang sederhana, dan tidak ada efek apapun pada pengembangan aplikasi. Kent Beck mengakui dan menegaskan bahwa XP tidak selalu cocok (sesuai) untuk setiap proyek pengembangan perangkat lunak. Namun, XP memiliki kelebihan yaitu sesuai untuk proyek yang memiliki *dynamic requirements* atau proyek yang memiliki berbagai *requirements* yang tidak jelas dari klien.

XP sangat cocok untuk pengembangan proyek yang memerlukan adaptasi cepat dalam perubahan-perubahan yang terjadi selama pengembangan aplikasi.



2.5.2 Kerangka Kerja eXtreme Programming

Tahapan pengembangan perangkat lunak dengan XP meliputi :

1. *Planning*/Perencanaan : Tahapan ini dimulai dengan pemahaman konteks bisnis dari aplikasi, mendefinisikan output, fitur yang ada pada aplikasi, fungsi dari aplikasi yang akan dibuat, serta alur pengembangan aplikasi.
2. *Design*/Perancangan : Tahapan ini menekankan pada desain aplikasi secara sederhana. Alat yang digunakan untuk mendesain pada tahap ini menggunakan kartu CRC (*Class Responsibility Collaborator*). CRC digunakan untuk pemetaan (membangun) kelas-kelas yang akan digunakan pada diagram *use case*, diagram kelas, dan diagram objek.
3. *Coding*/Pengkodean : Hal utama dalam pengembangan aplikasi dengan menggunakan XP adalah pair programming (dalam membuat program melibatkan 2 atau lebih programmer).
4. *Testing*/Pengujian : Tahapan ini berfokus pada pengujian fitur-fitur yang ada dalam aplikasi sehingga tidak terjadi kesalahan (*error*) dan aplikasi yang dibuat sesuai dengan kebutuhan pengguna.



2.6 Referensi Jurnal

NO	JUDUL	PENULIS/TAHUN	HASIL
1.	Sistem Pendukung Keputusan Seleksi Calon Karyawan dengan Metode PROMETHEE Studi Kasus Pamelia Group Yogyakarta	Azizah dan Winiarti (2014) Jurnal Sarjana Teknik Informatika, Vol. 2 Nomor 1, Februari 2014, e-ISSN: 2338-5197	Telah dibangun sistem penentuan seleksi calon karyawan pada Pamella Group, sehingga pihak HRD dapat mengambil keputusan dengan lebih baik
2.	Sistem Pendukung Keputusan Pengangkatan Karyawan Tetap Menggunakan Metode SAW (Simple Additive Weighting) Pada Industri Battery (AKI)	Nurma'ruf dan Herdi (2019), Jurnal Sistem Informasi dan E-Bisnis, Vol. 1, Issue 5, September 2019, P-ISSN: 2655-7541	Menghasilkan suatu perancangan sistem yang mempermudah proses pengangkatan karyawan tetap dibandingkan dengan sistem sebelumnya. Sistem dapat melakukan pengurutan prioritas proses produksi berdasarkan hasil perhitungan metode SAW dan sistem ini juga menghasilkan output berupa laporan-laporan yang berkaitan dengan laporan pengangkatan karyawan.



Lanjutan Tabel 2.6 Referensi Jurnal

3.	Aplikasi Sistem Pendukung Keputusan Pemilihan Karyawan Terbaik Menggunakan Metode Promethee pada Primkopti Jakarta Selatan	Mukhtar dan Adhy, Jurnal Masyarakat Informatika, Vol. 10, Nomor 2, ISSN: 2086-4930	Aplikasi ini menerapkan metode promethee dan nilai net flow dari perhitungan secara manual menunjukkan hasil yang sama. Kelebihan dari aplikasi ini yaitu kriteria dapat ditambahkan atau dikurangi sesuai dengan kebutuhan user.
4.	Implementasi Metode Topsis untuk Rekomendasi Pengangkatan Karyawan Tetap Pada PT. Bina Artha Ventura Cabang Nganjuk	Resti dan Ristyawan (2019), Jurnal Ilmiah Bidang Teknologi Informasi dan Komunikasi, Vol. 4, No.1, Januari 2019, P-ISSN : 2502-3470, E-ISSN : 2581-0367	Dalam penelitian ini menerapkan metode Topsis untuk menentukan kandidat terbaik yang dapat diangkat perusahaan menjadi karyawan tetap. Perusahaan menggunakan Metode Topsis untuk mengambil keputusan dengan waktu yang singkat dan biaya yang relatif rendah.



Lanjutan Tabel 2.6 Referensi Jurnal

5.	Simple Additive Weighting (SAW) Pada Sistem Pendukung Keputusan Untuk Pengangkatan Karyawan Tetap	Elfaladonna, dkk (2022), Jurnal Jaringan Sistem Informasi Robotik (JSR), Vol. 6, No. 1, Tahun 2022, E-ISSN: 2579-373X	Metode yang digunakan dalam penelitian ini adalah metode SAW. Aplikasi yang digunakan dapat mempermudah bagian personalia untuk menentukan karyawan tetap yang layak. Penulis mengharapkan adanya pengembangan aplikasi website yang lebih transparan dan adanya kolaborasi dengan metode lainnya dalam pengambilan keputusan agar lebih akurat.
----	---	---	--