

BAB II

TINJAUAN PUSTAKA

A. Landasan Teori

1. Penerapan

Pengertian penerapan adalah perbuatan menerapkan (Peter Salim dan Yenny Salim, 2002). Sedangkan menurut beberapa ahli berpendapat bahwa, penerapan adalah suatu perbuatan mempraktekkan suatu teori, metode, dan hal lain untuk mencapai tujuan tertentu dan untuk suatu kepentingan yang diinginkan oleh suatu kelompok atau golongan yang telah terencana dan tersusun sebelumnya. Berdasarkan pengertian tersebut dapat disimpulkan bahwa penerapan merupakan sebuah tindakan yang dilakukan baik secara individu maupun kelompok dengan maksud untuk mencapai tujuan yang telah dirumuskan. Adapun unsur-unsur penerapan meliputi:

- a. Adanya program yang dilaksanakan
- b. Adanya kelompok target, yaitu masyarakat yang menjadi sasaran dan diharapkan akan menerima manfaat dari program tersebut.
- c. Adanya pelaksanaan, baik organisasi atau perorangan yang bertanggung jawab dalam pengelolaan, pelaksanaan maupun pengawasan dari proses penerapan tersebut (Abdul Wahab, Solichin, 1990).

2. Sistem Pendukung Keputusan / *Decision Support System* (DSS)

a. Definisi Sistem Pendukung Keputusan

Menurut Nofriansyah & Defit (2017), Sistem Pendukung Keputusan adalah suatu informasi berbasis komputer yang menghasilkan sebagai alternatif keputusan untuk membantu manajemen dalam menangani berbagai permasalahan yang terstruktur maupun tidak terstruktur dengan menggunakan data dan model.

Decision Support System (DSS) biasanya dibangun untuk mendukung solusi atas suatu masalah atau untuk mengevaluasi suatu peluang. Aplikasi *Decision Support System* (DSS) digunakan dalam pengambilan keputusan. Aplikasi *Decision Support System* (DSS) menggunakan CBIS (Computer Based Information System) yang fleksibel, interaktif, dan dapat diadaptasi, yang dapat dikembangkan untuk mendukung solusi atas masalah manajemen spesifik yang tidak terstruktur.

Aplikasi *Decision Support System* (DSS) menggunakan data, memberikan antarmuka pengguna yang mudah, dan dapat menggabungkan pemikiran pengambilan keputusan. *Decision Support System* (DSS) lebih ditunjukkan untuk mendukung manajemen dalam melakukan pekerjaan yang bersifat analitis dalam situasi yang kurang terstruktur dan dengan kriteria yang kurang jelas. *Decision Support System* (DSS) tidak dimaksudkan untuk mengotomatiskan pengambilan keputusan, tetapi memberikan perangkat interaktif yang memungkinkan

pengambilan keputusan, tetapi memberikan perangkat interaktif yang memungkinkan pengambilan keputusan untuk melakukan berbagai analisis menggunakan model-model yang tersedia.

Sistem Pendukung Keputusan terdapat beberapa jenis solusi pemecahan masalah diantaranya yaitu:

- 1) *Multi Attribute Decision Making* (MADM)
 - a) Metode *Simple Additive Weighting* (SAW)
 - b) Metode *Weight Product* (WP)
 - c) Metode *Analytical Hierarchy Process* (AHP)
 - d) Metode *Topsis* dan lain-lain.
- 2) *Multi Criteria Decision Making* (MCDM)
 - a) Metode *Promethee*
 - b) Metode *Electre*
 - c) Metode *Oreste*
 - d) Metode *Entropi* dan lain-lain.
- 3) *Multi Factor Evaluation Process* (MFEP)
- 4) *Fuzzy Multi Attribute Decision Making* (FMADM)
 - a) F-AHP
 - b) F-SAW

b. **Komponen Sistem Pendukung Keputusan**

Menurut Delen & Turban (2014) untuk dapat menerapkan Sistem Pendukung Keputusan terdapat 4 komponen subsistem yang harus disediakan yaitu:

1) Subsistem Manajemen Data

Subsistem ini menyediakan data bagi sistem, termasuk didalamnya basisdata. Berisi data yang relevan untuk situasi dan diatur oleh perangkat lunak yang disebut *Database Management System* (DBMS).

2) Subsistem Manajemen Model

Subsistem ini berfungsi sebagai pengelola berbagai model, mulai dari model keuangan, statistik, matematik, atau model kuantitatif lainnya yang memiliki kemampuan analisis dan manajemen perangkat lunak yang sesuai. Perangkat lunak ini sering disebut *Model Base Management System* (MBMS).

3) Subsistem Manajemen Pengetahuan

Subsistem ini mendukung berbagai subsistem lainnya, atau dapat dikatakan berperan sebagai komponen yang independen. Subsistem ini menyediakan intelegensi untuk menambah pertimbangan pengambil keputusan.

4) Subsistem Manajemen Antar Muka Pengguna

Subsistem ini berupa tampilan yang disediakan yang mampu mengintegrasikan sistem terpasang dengan pengguna secara interaktif. Melalui subsistem ini pengguna dapat berkomunikasi dengan sistem pendukung keputusan serta memerintah sistem pendukung keputusan.

c. Tujuan Sistem Pendukung Keputusan

Menurut Indriantoro & Utami, (2016) Tujuan Sistem Pendukung Keputusan (SPK) yaitu:

- 1) Membantu manajemen membuat keputusan untuk memecahkan masalah semi terstruktur.
- 2) Mendukung penilaian manajer bukan mencoba menggantikannya.
- 3) Meningkatkan efektifitas pengambilan keputusan manajer daripada efisiensinya.

d. Karakteristik Sistem Pendukung Keputusan

Menurut Indriantoro & Utami, (2016) Sistem Pendukung Keputusan memiliki beberapa karakteristik sebagai berikut:

- 1) Tujuan utama dari Sistem Pendukung Keputusan adalah untuk memperbaiki mutu keputusan serta *performance*. Sistem Pendukung Keputusan tidak hanya sekedar menyajikan informasi yang lebih banyak, lebih baik dan lebih akurat pada waktu yang tepat saja.
- 2) Sistem Pendukung Keputusan ditujukan untuk environment yang kompleks, kurang terstruktur dan bahkan politis sifatnya.
- 3) Sistem Pendukung Keputusan bertumpu pada laporan perkecualian dan macamnya untuk menunjang proses identifikasi masalah.
- 4) Sistem Pendukung Keputusan berkombinasi “modelling” dan teknik-teknik analisa yang lain dengan fungsi penyajian kembali data.

5) Sistem Pendukung Keputusan berfokus pada prinsip “mudah dipakai” dan “fleksibel” dalam berhadapan dengan pemakai tertentu atau sekelompok pemakai.

6) Proses Pengambilan Keputusan.

e. Manfaat Sistem Pendukung Keputusan

Sistem Pendukung Keputusan dapat memberikan berbagai macam manfaat. Menurut Laksana & Zarkasy, (2015) manfaat yang dapat diambil dari DSS adalah:

- 1) SPK memperluas kemampuan pengambil keputusan dalam memproses data / informasi bagi pemakainya.
- 2) SPK membantu pengambil keputusan untuk memecahkan masalah terutama berbagai masalah yang sangat kompleks dan tidak terstruktur.
- 3) SPK dapat menghasilkan solusi dengan lebih cepat serta hasilnya dapat diandalkan.
- 4) Walaupun suatu SPK, mungkin saja tidak mampu memecahkan masalah yang dihadapi oleh pengambil keputusan, namun ia dapat menjadi stimulan bagi pengambil keputusan dalam memahami persoalannya, karena mampu menyajikan berbagai alternatif pemecahan.

3. Dosen

Dosen adalah pendidik profesional dan ilmuwan, yakni seseorang yang memiliki kapabilitas dalam bidang keilmuan tertentu, dan dengan

kapabilitasnya itu, dia bisa mengajar, meneliti untuk mengembangkan teori-teori serta teknologi dalam bidang keilmuannya, dan bahkan melakukan pengabdian pada masyarakat untuk memvalidasi teori atau teknologi yang sudah dihasilkan lewat penelitiannya. Seorang dosen harus memiliki kapabilitas yang baik dalam bidang keilmuannya, dan itu ditandai dengan pendidikan yang linier dengan cabang atau bidang ilmu yang akan menjadi tanggung jawabnya, kemudian, produktif dalam menulis paper dalam bidang ilmunya itu, baik untuk bahan ajar maupun paper untuk disampaikan dalam forum seminar atau simposium, dan bahkan pada jenjang kepangkatan untuk memperoleh Guru Besar (Rosyada, 2015).

4. *Preference Ranking Organization Method for Enrichment Evaluations (PROMETHEE)*

PROMETHEE (Preference Ranking Organization Method for Enrichment Evaluations) merupakan salah satu metode dari MCDM (*Multi Criteria Decision Making*) untuk penentuan urutan (prioritas) dalam analisis. Masalah pokoknya adalah kesederhanaan, kejelasan, dan kestabilan. Sangat tepat untuk digunakan karena dugaan dari dominasi kriteria yang digunakan dalam promethee adalah penggunaan nilai dalam hubungan *outranking*. Sehingga diperoleh solusi atau hasil dari beberapa alternatif untuk diambil sebuah keputusan (Indriantoro & Utami, 2016).

Promethee mempunyai kemampuan untuk menangani bayak perbandingan, pengambil keputusan hanya mendefinisikan skala ukurannya sendiri tanpa batasan, untuk mendefinisikan prioritasnya dan preferensi

untuk setiap kriteria dengan memutuskan pada nilai (value), tanpa memikirkan tentang metode perhitungan.

Metode *promethee* menggunakan kriteria dan bobot dari masing-masing kriterianya yang kemudian diolah untuk menentukan pemilihan alternatif lapangan, yang hasilnya berurutan berdasarkan prioritasnya. Penggunaan metode *promethee* dapat dijadikan metode untuk pengambilan keputusan di bidang pemasaran, sumber daya manusia, pemilihan lokasi, atau bidang lain yang berhubungan dengan pemilihan alternatif.

Prinsip yang digunakan adalah penetapan prioritas alternatif yang telah ditetapkan berdasarkan pertimbangan ($\forall_i | f_i(.) \in \mathcal{R}$) dengan kaidah dasar:

$$\text{Max } \{f_1(x), f_2(x), f_3(x), \dots, f_k(x) | x \in \mathcal{R}\}$$

Di mana K adalah sejumlah kumpulan alternatif, $f_i = (i = 1, 2, 3, 4, 5, \dots, K)$ merupakan nilai/ukuran relatif kriteria untuk masing-masing alternatif. Dalam aplikasinya sejumlah kriteria telah ditetapkan untuk menjelaskan K yang merupakan penilaian dari $\mathcal{R}(\text{Real})$.

Promethee termasuk dalam keluarga metode outranking yang dikembangkan oleh B.Roy (1985) dan meliputi dua fase:

1. Membangun hubungan *outranking* dari K
2. Eksploitasi dari hubungan ini memberikan jawaban optimasi \mathcal{R} kriteria dalam paradigm permasalahan multikriteria.

Dalam fase pertama, nilai hubungan *outranking* berdasarkan pertimbangan dominasi masing-masing kriteria indeks preferensi ditentukan

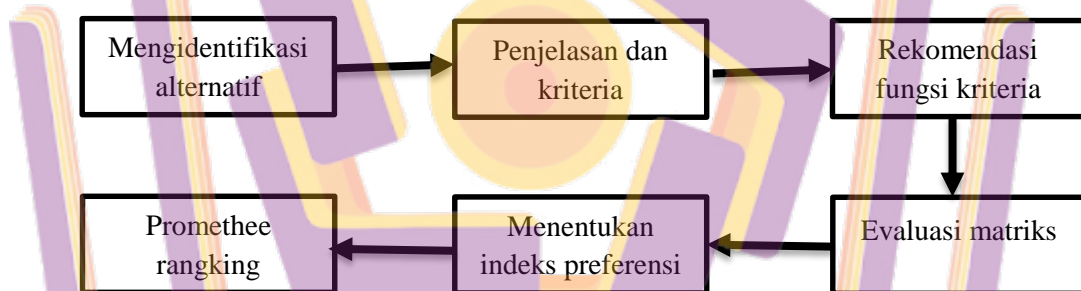
dan nilai *outranking* secara grafis disajikan berdasarkan preferensi dari pembuat keputusan. Data dasarnya disajikan sebagai berikut:

Tabel 2.1 Data Dasar Analisis Promethee

Alternatif	$f_1(.)$	$f_2(.)$...	$f_j(.)$...	$f_k(.)$
a1	$f_1(a_1)$	$f_2(a_1)$...	$f_j(a_1)$...	$f_k(a_1)$
a2	$f_1(a_2)$	$f_2(a_2)$...	$f_j(a_2)$...	$f_k(a_2)$
...
a_i	$f_1(a_i)$	$f_2(a_i)$...	$f_j(a_i)$...	$f_k(a_i)$
...
a_n	$f_1(a_n)$	$f_2(a_n)$...	$f_j(a_n)$...	$f_k(a_n)$

Sumber: Suryadi dan Ramadhani, 1998

a. Proses Promethee



Gambar 2.1 Diagram Proses Promethee (Suryadi dan Ramdhani, 2002)

Pada Gambar 2.1 merupakan langkah-langkah yang digunakan metode ini adalah sebagai berikut (Suryadi dan Ramdhani, 2002):

- 1) Mengidentifikasi alternatif.
- 2) Penjelasan dari kriteria, alternatif (a) dievaluasi dari beberapa kriteria (k), yang harus dimaksimalkan atau diminimalkan.
- 3) Rekomendasi fungsi preferensi untuk keperluan aplikasi. Dalam Promethee disajikan enam fungsi kriteria, hal ini tentu saja tidak mutlak, tetapi bentuk ini cukup baik untuk beberapa kasus.

4) Evaluasi matrik, saat kriteria dan alternatif sudah terpilih, langkah selanjutnya adalah membuat matrik *payoff*. Table matrik ini untuk setiap pasangan kriteria-kriteria, ukuran kuantitatif dan kualitatif dari efek yang dihasilkan oleh alternatif berhubungan dengan kriteria tersebut. Suatu matrik dapat terdiri dari data ukuran kardinal atau skala ordinal.

5) Menentukan indeks prefensi multikriteria. Preferensi dinyatakan dengan angka antara 0 dan 1, dan dinilai dengan prosedur tertentu.

6) *Promethee ranking*. Arah dalam grafik nilai *outranking* ditentukan berdasarkan *Leaving flow*, *Entering flow*, dan *Leaving flow* adalah jumlah nilai garis lengkung yang memiliki arah menjauh dari *node a* dan hal ini merupakan karakter pengukuran *outranking*. Penjelasan dari hubungan *outranking* dibangun atas pertimbangan untuk aksi pada grafik nilai *outranking*, berupa urutan parsial (*Promethee*) dan urutan lengkap (*Promethee*) pada sejumlah aksi yang mungkin, yang dapat diusulkan pada pembuat keputusan untuk memperkaya penyelesaian masalah karakteristik data.

b. Dominasi Kriteria

Nilai f merupakan nilai nyata dari suatu kriteria:

$$f_i K \rightarrow \mathfrak{R}$$

dan tujuan berupa prosedur optimasi. Untuk setiap alternatif $a \in K, f(a)$ merupakan evaluasi dari alternatif tersebut untuk suatu kriteria. Pada saata dua alternatif dibandingkan, $a, b \in K$, harus dapat ditentukan perbandingan preferensinya. Penyampaian intensitas (P) dari preferensi alternatif a terhadap alternatif b sedemikian rupa sehingga:

- 1) $P(a,b) = 0$, menunjukkan tidak ada perbedaan antara alternatif a dan alternatif b berdasarkan semua kriteria.
- 2) $P(a,b) \sim 0$, menunjukkan preferensi yang lemah untuk alternatif a > alternatif b berdasarkan semua kriteria.
- 3) $P(a,b) \sim 1$, menunjukkan preferensi yang kuat untuk alternatif a > alternatif b berdasarkan semua kriteria.
- 4) $P(a,b) = 1$, berarti mutlak preferensi dari a lebih baik dari b.

Dalam metode ini, fungsi preferensi seringkali menghasilkan nilai fungsi yang berbeda antara dua evaluasi, sehingga:

$$A = P(A,B) = P\{f(A) - f(B)\}$$

Keterangan:

A = alternatif A B = alternatif B

$P(A,B)$ = preference index alternatif A terhadap alternatif B $f(A)$ = nilai fungsi alternatif A

$f(B)$ = nilai fungsi alternatif B

untuk semua kriteria, suatu alternatif akan dipertimbangkan memiliki nilai kriteria yang lebih baik ditentukan oleh nilai f dan akumulasi dari

nilai ini menentukan nilai preferensi atas masing-masing alternative yang kan dipilih.

c. Rekomendasi Fungsi Preferensi untuk Keperluan Aplikasi

Dalam *promethee* disajikan enam bentuk fungsi preferensi kriteria, yaitu kriteria biasa (*usual criterion*), kriteria Quasi (*Quasi criterion*), kriteria dengan preferensi linier (*U-shape criterion*), kriteria dengan preferensi linier dan area yang tidak berbeda (*V-shapecriterion*), kriteria Gaussian (*Gaussian Criterion*). Hal ini tentu saja tidak mutlak, tetapi bentuk ini cukup baik untuk beberapa kasus.

Untuk memeberikan gambaran yang baik terhadap area yang tidak sama, digunakan fungsi selisih nilai kriteria anantara alternatif H (d) dimana hal ini mempunyai hubungan langsung padafungsi preferensi P:

$$\left. \begin{array}{l} a, b \\ f(a), f(b) \end{array} \right\} \begin{array}{l} f(a) > f(b) \text{ a P b} \\ f(a) > f(b) \text{ a I b} \end{array}$$

Keterangan:

a = alternatif a b = alternatif b

f(a) = niali fungsi alternatif a f(b) = nilai fungsi alternatif b

a P b = alternatif a outrank alternatif b

a I b = alternatif a tidak beda dengan alternatif b

d. Rekomendasi Fungsi Preferensi

Dalam *promethee* disajikan enam bentuk fungsi preferensi kriteria. Enam preferensi tersebut adalah sebagai berikut:

1) Kriteria Biasa (*Usual Criterion*)

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

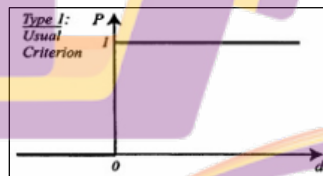
Keterangan:

a) $H(d)$ = fungsi selisih kriteria antara alternatif

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

Tipe *usual* adalah tipe dasar, yang tidak memiliki nilai *threshold* atau kecenderungan dan tipe ini jarang digunakan. Pada kasus ini, tidak ada beda (sama penting) antara a dan b jika dan hanya jika $f(a) = f(b)$: apabila nilai kriteria pada masing-masing alternatif memiliki nilai berbeda, pembuat keputusan membuat preferensi mutlak untuk alternatif memiliki nilai yang lebih baik.

Fungsi $H(d)$ untuk preferensi disajikan pada Gambar 2.1



Gambar 2.2 Usual Criterion

(Sumber: J.P. Brans, dan B.Mareschal, 1997)

2) Kriteria Quasi (*Quasi Criterion*)

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

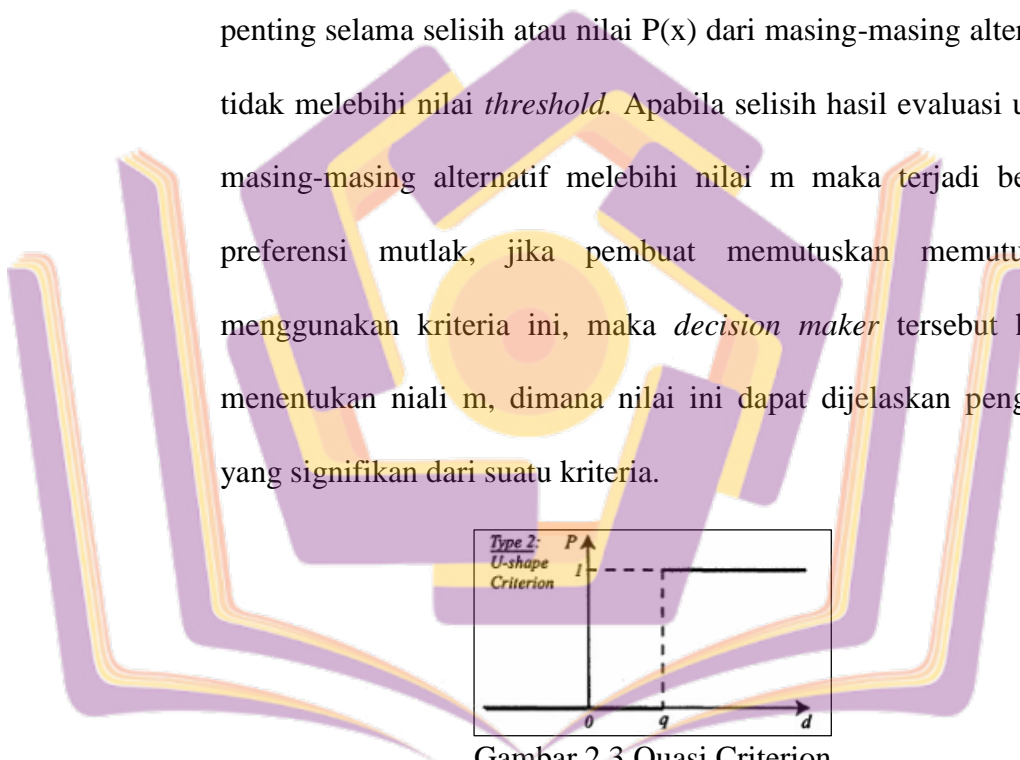
Keterangan:

a) $H(d)$: Fungsi selisih kriteria antar alternative

b) d : Selisih nilai kriteria $\{d = f(a) - f(b)\}$

c) Parameter (q) : Harus merupakan nilai yang tetap

Tipe quasi sering digunakan dalam penilaian suatu data dari segi kualitas atau mutu, yang mana tipe ini menggunakan satu *threshold* atau kecenderungan yang sudah ditentukan, dalam kasus ini *threshold* itu adalah *indifference*. *Indifference* ini biasanya dilambangkan dengan karakter m atau q , dan nilai *indifference* harus diatas 0 (nol). Suatu alternatif memiliki nilai preferensi yang sama penting selama selisih atau nilai $P(x)$ dari masing-masing alternatif tidak melebihi nilai *threshold*. Apabila selisih hasil evaluasi untuk masing-masing alternatif melebihi nilai m maka terjadi bentuk preferensi mutlak, jika pembuat memutuskan menggunakan kriteria ini, maka *decision maker* tersebut harus menentukan nilai m , dimana nilai ini dapat dijelaskan pengaruh yang signifikan dari suatu kriteria.



Gambar 2.3 Quasi Criterion
(Sumber: J.P. Brans, dan B.Mareschal, 1997)

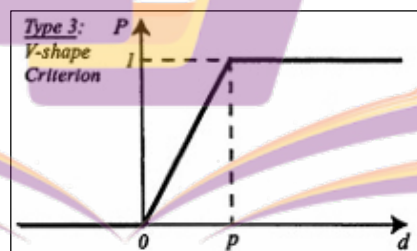
3) Kriteria dengan Preferensi Linier

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

Keterangan:

- a) $H(d)$: Fungsi selisih kriteria antar alternative
- b) d : Selisih nilai kriteria $\{d = f(a) - f(b)\}$
- c) p : Nilai kecenderungan atas

Tipe linier sering digunakan dalam penilaian dari segi kuantitatif atau banyaknya jumlah, yang mana tipe ini juga menggunakan satu *threshold* atau kecenderungan yang sudah ditentukan, dalam kasus ini *threshold* itu adalah *preference*. 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, sesuai dengan persamaan.



Gambar 2.4 Kriteria Preferensi Linier
(Sumber: J.P. Brans, dan B.Mareschal, 1997)

Pada saat pembuat keputusan mengidentifikasi beberapa kriteria untuk tipe ini, harus ditentukan nilai dari kecenderungan atas (nilai p). Dalam hal ini, nilai d diatas p telah dipertimbangkan akan memberikan preferensi mutlak dari suatu alternatif.

Pada saat pembuat keputusan mengidentifikasi beberapa kriteria untuk tipe ini, harus menentukan nilai dari kecenderungan atas (nilai p). Dalam hal ini, nilai d diatas p telah dipertimbangkan akan memberikan preferensi mutlak dari satu alternatif.

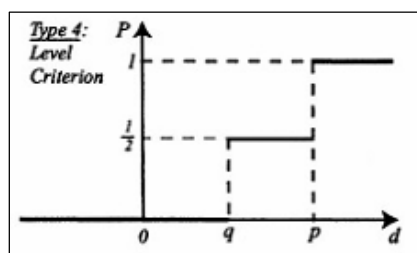
4) Kriteria *Level* (*Level Criteria*)

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

Keterangan:

- $H(d)$: Fungsi selisih kriteria antar alternative
- p : Nilai kecenderungan atas
- Parameter (q) : Harus merupakan nilai yang tetap.

Tipe ini mirip dengan tipe quasi yang sering digunakan dalam penilaian suatu data dari segi kualitas atau mutu. Dalam kasus ini, kecenderungan tidak berbeda q dan kecenderungan preferensi p adalah ditentukan secara simultan. Jika d berada diantara nilai q dan p , hal ini berarti situasi preferensi yang lemah ($H(d) = 0,5$). Pembuat keputusan telah menentukan kedua kecenderungan untuk kriteria ini.



Gambar 2.5 Level Criterion
(Sumber: J.P. Brans, dan B.Mareschal, 1997)

5) Kriteria dengan Preferensi Linier dan Area yang Tidak Berbeda

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

Keterangan:

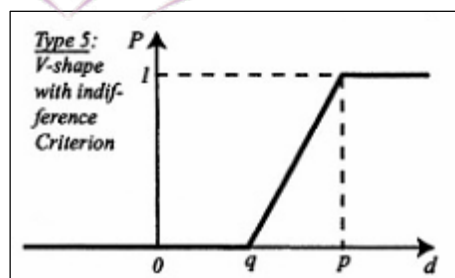
a) $H(d)$: Fungsi selisih kriteria antar alternative

b) d : Selisih nilai Kriteria $\{d=f(a) - f(b)\}$

c) Parameter (p) : nilai kecenderungan atas

d) Parameter (q) : Harus merupakan nilai yang

Tipe linear dan area yang tidak berbeda juga mirip dengan tipe linear yang sering digunakan dalam penilaian dari segi kuantitatif atau banyaknya jumlah. Pada kasus ini, pengambil keputusan mempertimbangkan peningkatan preferensi secara linier dari tidak berbeda hingga preferensi mutlak dalam area antara dua kecenderungan q dan p . Dimana dua parameter tersebut sudah ditentukan sebelumnya.



Gambar 2.6 Kriteria dengan Preferensi Linier dan Area Yang Tidak Berbeda (Sumber: J.P. Brans, dan B.Mareschal, 1997)

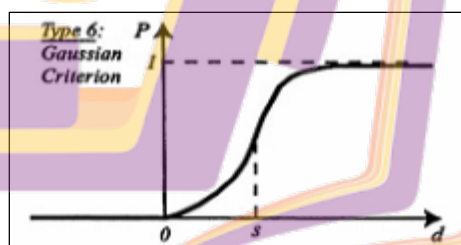
6) Kriteria Gaussian (*Gaussian Criterion*)

$$H(d) = \begin{cases} 0 & \text{jika } d \leq 0 \\ 1 - \exp\left(-\frac{d^2}{2a^2}\right) & \text{jika } d > 0 \end{cases}$$

Keterangan:

a = kriteria Gaussian

Tipe gaussian sering digunakan untuk mencari nilai aman atau titik aman pada data yang bersifat *continue* atau berjalan terus. Fungsi ini bersyarat apabila telah ditentukan nilai, dimana dapat dibuat berdasarkan distribusi normal dalam statistik atau nilai standar deviasi. Pengambilan keputusan mempertimbangkan peningkatan preferensi secara linier dari tidak berbeda hingga preferensi mutlak dalam area antara dua kecenderungan m dan n .



Gambar 2.7 Gaussian Criterion
(Sumber: J.P. Brans, dan B.Mareschal, 1997)

e. Nilai *threshold* atau kecenderungan

Enam tipe dari penyamarataan kriteria bisa dipertimbangkan dalam metode *Promethee*, tiap-tiap tipe bisa lebih mudah ditentukan nilai kecenderungannya atau parameternya karena hanya satu atau dua parameter yang mesti ditentukan. Hanya tipe *usual* (biasa) saja yang tidak memiliki nilai parameter.

1) *Indifference threshold* yang biasa dilambangkan dalam karakter m atau q . Jika nilai perbedaan (x) di bawah atau sama dengan nilai $indifference\ x \leq m$ maka x dianggap tidak memiliki nilai perbedaan $= 0$.

2) *Preference threshold* yang biasa dilambangkan dalam karakter n atau p . Jika nilai perbedaan (x) di atas atau sama dengan nilai $preferencex \geq n$ maka perbedaan tersebut memiliki nilai mutlak $x = 1$.

3) *Gaussian threshold* yang biasa dilambangkan dalam karakter σ serta diketahui dengan baik sebagai parameter yang secara langsung berhubungan dengan nilai standar deviasi pada distribusi normal.

f. Indeks Preferensi Multikriteria

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

Indeks preferensi multi kriteria ditentukan berdasarkan rata-rata bobot dari fungsi preferensi P_i .

$$\varphi(a + b) \sum_{i=1}^n \pi_i P_i(a, b) \sqrt{a}, b \in A$$

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

- 1) $\Phi(a,b) = 0$ menunjukkan preferensi yang lemah untuk alternatif a > alternatif b berdasarkan semua kriteria.
- 2) $\Phi(a,b) = 1$ menunjukkan preferensi yang kuat untuk alternatif a > alternatif b berdasarkan semua kriteria.

Indeks preferensi ditentukan berdasarkan nilai hubungan *outranking* pada sejumlah kriteria dari masing-masing alternatif. Hubungan ini dapat disajikan sebagai grafik nilai *outranking*, node-nodenya merupakan alternatif berdasarkan penilaian kriteria tertentu. Diantara dua node (alternatif), a dan b, merupakan garis lengkung yang mempunyai nilai $\varphi(b,a)$ dan $\varphi(a,b)$ (tidak ada hubungan khusus antara $\varphi(b,a)$ dan $\varphi(a,b)$).

g. *Promethe Rangking*

Perhitungan arah preferensi dipertimbangkan berdasarkan nilai indeks:

1) *Leaving Flow*

Jumlah dari yang memiliki arah menjauh dari *node* a. dan hal ini merupakan pengukuran *outranking*. Untuk setiap *node* a dalam

grafik nilai *outranking* ditentukan berdasarkan *leaving flow*, dengan persamaan:

$$\emptyset^+(a) = \frac{1}{n-1} \sum_{\varphi(a,x)} \varphi(a,x)$$

dimana $\varphi(a,x)$ menunjukkan preferensi bahwa alternatif a lebih baik dari alternatif x.

2) *Entering flow*

Entering flow adalah jumlah dari yang memiliki arah mendekat dari *node* a dan hal ini merupakan karakter pengukuran *outranking*. *Entering flow* diukur berdasarkan karakter *outranked* dari a.

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

3) *Net flow*

Net flow diukur dengan menghitung selisih *leaving flow* dan *entering flow*.

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

Keterangan:

- a) $\varphi(a,x)$ = menunjukkan preferensi bahwa alternatif a lebih baik dari alternatif x.
- b) $\varphi(x,b)$ = menunjukkan preferensi bahwa alternatif x lebih baik dari alternatif a.

- c) $\phi^+(a) = \text{Leaving flow}$, digunakan untuk menentukan urutan prioritas pada proses *Promethee I* yang menggunakan urutan parsial.
- d) $\phi^-(a) = \text{Entering flow}$, digunakan untuk menentukan urutan prioritas pada proses *Promethee I* yang menggunakan urutan parsial.
- e) $\phi(a) = \text{Net flow}$, digunakan untuk menghasilkan keputusan akhir penentuan urutan dalam menyelesaikan masalah sehingga menghasilkan urutan lengkap.

Penjelasan dari hubungan *outranking* dibangun atas pertimbangan untuk masing-masing alternatif pada grafik nilai *outranking*, berupa urutan parsial (*Promethee I*) atau urutan lengkap (*Promethee II*) pada sejumlah alternatif yang mungkin, yang dapat diusulkan kepada pembuat keputusan untuk memperkaya penyelesaian masalah.

h. *Promethee I*

Nilai terbesar pada *leaving flow* dan nilai yang kecil dari *entering flow* merupakan alternatif yang terbaik. *Leaving flow* dan *entering flow* menyebabkan:

$$\begin{cases} a P^+ b & \text{jika } \phi^+(a) > \phi^+(b) \\ a I^+ b & \text{jika } \phi^+(a) = \phi^+(b) \end{cases}$$

$$\begin{cases} a P^- b & \text{jika } \phi^-(a) > \phi^-(b) \\ a I^- b & \text{jika } \phi^-(a) = \phi^-(b) \end{cases}$$

Keterangan:

$a P^+ b$ = nilai *leaving flow* alternatif a *outrank* alternatif b

$a I^+ b$ = nilai *leaving flow* alternatif a tidak beda alternatif b

$a P^- b$ = nilai *entering flow* alternatif a *outrank* alternatif b

$a I^- b$ = nilai *entering flow* alternatif a tidak beda alternatif b

$\phi^+(a)$ = *Leaving flow* alternatif a, digunakan untuk menentukan urutan prioritas pada proses *Promethee I* yang menggunakan urutan parsial.

$\phi^+(b)$ = *Leaving flow* alternatif b, digunakan untuk menentukan urutan prioritas pada proses *Promethee I* yang menggunakan urutan parsial.

$\phi^-(a)$ = *Entering flow* alternatif a, digunakan untuk menentukan urutan prioritas pada proses *Promethee I* yang menggunakan urutan parsial.

$\phi^-(b)$ = *Entering flow* alternatif b, digunakan untuk menentukan urutan prioritas pada proses *Promethee I* yang menggunakan urutan parsial.

Promethee I menampilkan *partial preorder* (PI, II, RI) dengan mempertimbangkan interseksi dari dua *preorder*:

$a P_i b$ (a outrank b) jika $a P^+ b$ dan $a P^- b$ @ atau $a P^+ b$ d

Keterangan:

$a P_I b$ = *partial preorder* yang menunjukkan alternatif a *outrank* alternatif b

$a P_{II} b$ = *partial preorder* yang menunjukkan alternatif a tidak beda alternatif b

$a P_{RI} b$ = *partial preorder* yang menunjukkan alternatif a *incomparable* dengan alternatif b

Partial preorder diajukan kepada pembuat keputusan, untuk membantu pengambilan keputusan masalah yang dihadapinya. Dengan menggunakan metode *Promethee I* masih menyisakan bentuk *incomparable*, atau dengan kata hanya memberikan solusi *partial preorder* (sebagian).

i. *Promethee II*

Dalam kasus *complete preorder* dalam K adalah penghindaran dari bentuk *incomparable*, *Promethee II complete preorder* (PII, III) disajikan dalam bentuk *net flow* disajikan berdasarkan pertimbangan persamaan:

$$\begin{cases} a \text{ PII } b \text{ jika } \varphi(a) > \varphi(b) \\ a \text{ III } b \text{ jika } \varphi(a) = \varphi(b) \end{cases}$$

Keterangan:

$a \text{ PII } b$ = *complete preorder* yang menunjukkan alternatif a outrank alternatif b

$a \text{ III } b$ = *complete preorder* yang menunjukkan alternatif a tidak beda alternatif b

$\varphi(a)$ = *Net flow* alternatif a, digunakan untuk menghasilkan keputusan akhir penentuan urutan dalam menyelesaikan masalah sehingga menghasilkan urutan lengkap.

$\varphi(b)$ = *Net flow* alternatif b, digunakan untuk menghasilkan keputusan akhir penentuan urutan dalam menyelesaikan masalah sehingga menghasilkan urutan lengkap.

5. Website

a. Definisi Website

Website adalah kumpulan dari halaman *web* yang sudah dipublikasikan di jaringan internet dan memiliki *domain* atau URL (Uniform Resource Locator) yang dapat diakses semua pengguna internet dengan cara mengentikan alamatnya. Hal ini dimungkinkan dengan adanya teknologi World Wide Web (WWW) fasilitas hypertext guna menampilkan data berupa teks, gambar, animasi, suara dan multimedia lainnya (Arief, 2011).

Website adalah keseluruhan halaman-halaman *web* yang terdapat dalam sebuah *domain* yang mengandung informasi. Sebuah *website* biasanya dibangun atas banyak halaman *web* yang saling berhubungan. Jadi dapat dikatakan bahwa, pengertian *website* adalah kumpulan halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara, dan atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing dihubungkan dengan jaringan-jaringan halaman. Hubungan antara satu halaman website dengan halaman website lainnya disebut dengan *hyperlink*, sedangkan teks yang dijadikan media penghubung disebut *hypertext* (Hidayat, 2010).

b. Jenis-jenis *website*

Terdapat beberapa jenis *website* yang berdasarkan pada sifatnya atau *style*-nya (Prasetyo, 2015) yaitu:

1) *Website* Dinamis

Website dinamis merupakan sebuah *website* yang menyediakan *content* atau isi yang selalu berubah-ubah setiap saat. Bahasa pemrograman antara lain PHP, ASP, .NET dan memanfaatkan *database* MySQL.

2) *Website* Statis

Website statis merupakan *website* yang *content*-nya sangat jarang diubah. Bahasa pemrograman yang digunakan adalah HTML dan belum memanfaatkan *database*.

6. *Codeigniter*

Codeigniter adalah sebuah *framework* PHP yang dapat membantu mempercepat *developer* dalam pengembangan aplikasi *website* berbasis PHP dibandingkan jika menulis semua kode program dari awal (Basuki, 2010).

Codeigniter adalah salah satu *framework* PHP bahkan *framework* PHP yang paling *powerfull* saat ini, karena di dalamnya terdapat fitur lengkap aplikasi *web* dimana fitur-fitur tersebut sudah dikemas menjadi satu. Selain itu, *Codeigniter* juga saat ini banyak digunakan khususnya bagi *developer web* untuk mengembangkan aplikasi berbasis webnya (Hidayatullah dan Kawistara, 2017).

7. Model View Controller (MVC)

Model View Controller menurut (Hidayatullah dan Kawistara, 2010) atau yang kita sebut dengan MVC adalah sebuah metode yang memisahkan data *logic (Model)* dari *presentation logic (View)* dan *process logic (Controller)* atau secara sederhana adalah memisahkan antara desain interface, data, dan process. Dalam metode MVC terdapat tiga komponen yaitu:

1) Model

Model bertugas untuk mengelola *database*, model akan berisi *class* ataupun fungsi serta perintah-perintah *query* untuk membuat (*create*), pembaruan (*update*), menghapus (*delete*), mencari data (*search*) dan mengambil data (*select*) pada database.

2) View

View adalah bagian *user interface* atau bagian yang nantinya merupakan tampilan untuk *end-user*. Dalam *view* tidak boleh terdapat pemrosesan data ataupun pengaksesan yang berhubungan dengan data.

View hanya menampilkan data-data hasil dari model dan *controller*, biasanya berupa halaman html, css, javascript, ajax, dan lain-lain.

3) Controller

Controller adalah penghubung antara *view* dan *model*, *Controller* bertugas sebagai pemrosesan data alur *logic program*, menyediakan *variable* yang akan ditampilkan di *view*, pemanggilan

model sehingga *model* dapat mengakses database, *error handling*, validasi atau *check* terhadap suatu inputan.

8. Basis Data

Basis data adalah kumpulan yang terorganisasi dari data-data yang berhubungan dengan sedemikian rupa sehingga mudah disimpan, dimanipulasi serta dipanggil oleh pengguna (Nugroho, 2011). Pendapat lain menyatakan bahwa basis data adalah koleksi dari data-data yang terorganisasi dengan cara sedemikian rupa sehingga mudah dalam disimpan dan dimanipulasi (diperbaharui, dicari, diolah dengan perhitungan-perhitungan tertentu, serta dihapus) (Nugroho, 2004).

Menurut Fathansyah (2012), basis data terdiri dari 2 kata, yaitu basis dan data. Basis kurang lebih dapat diartikan sebagai markas atau gudang, tempat bersarang atau berkumpul. Sedangkan data adalah representasi fakta dunia nyata yang mewakili suatu objek seperti manusia, barang, hewan, peristiwa, konsep dan sebagainya.

9. *Unified Modeling Language* (UML)

Unified Modeling Language (UML) adalah keluarga notasi grafis yang didukung oleh meta-model tunggal, yang membantu pendeskripsian dan desain sistem perangkat lunak, khususnya sistem yang dibangun menggunakan pemrograman berorientasi objek. (Fowler, 2005).

Menurut Yuni Sugiarti (2013), UML di definisikan sebagai bahasa visual untuk menjelaskan, memberi spesifikasi, merancang, membuat model, dan mendokumentasikan aspek-aspek dari sebuah sistem.

Dari pengertian di atas dapat disimpulkan bahwa UML adalah bahasa pemodelan untuk membangun perangkat lunak yang dibangun menggunakan teknik pemrograman berorientasi objek untuk menjelaskan, memberi spesifikasi, merancang, membuat model dan mendokumentasikan aspek-aspek dari sebuah sistem.

a. *Class diagram*

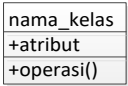





Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem.

Kelas memiliki apa yang disebut atribut dan metode atau operasi.

- 1) Atribut merupakan variable-variabel yang dimiliki oleh suatu kelas.
- 2) Atribut mendeskripsikan properti dengan sebaris teks didalam kotak kelas tersebut.
- 3) Operasi atau metode adalah fungsi-fungsi yang dimiliki oleh satu kelas.

Class diagram mendeskripsikan jenis-jenis objek dalam sistem dan berbagai hubungan statis yang terdapat diantara mereka. Diagram kelas juga menunjukkan properti dan operasi sebuah kelas dan batasan-batasan yang terdapat dalam hubungan-hubungan objek tersebut.

Tabel 2.2 Simbol *Class Diagram*

No	Nama	Simbol	Keterangan
1	Operasi		Kelas pada struktur sistem.
2	Antarmuka/ <i>interface</i>		Sama dengan konsep interface dalam pemrograman berorientasi objek.
3	Asosiasi berarah		Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain.
4	Generalisasi		Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus)
5	Kebergantungan		Relasi antar tabel dengan makna kebergantungan antar kelas
6	Agresi		Relasi antar kelas dengan makna semua bagian


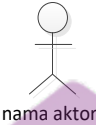


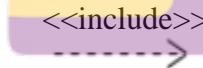
(Yuni Sugiarti,2013)

b. *Use Case Diagram*

Use case diagram merupakan pemodelan untuk menggambarkan kelakuan sistem dibuat. *Use case diagram* mendeskripsikan sebuah interaksi antar satu atau lebih aktor dengan sistem yang dibuat. dengan pengertian cepat, *Use case diagram* digunakan untuk mengetahui fungsi apa saja yang ada didalam sebuah sistem dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Terdapat beberapa simbol dalam menggambarkan diagram *use case*, aktor dan relasi.

Berikut adalah simbol-simbol yang adapada *use case* :

Table 2.3 Simbol *Use Case Diagram*

No	Nama	Simbol	Keterangan
1	<i>Use case</i>		Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.
2	Aktor/ <i>actor</i>		Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri.
3	Asosiasi/ <i>association</i>		Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
4	Extend		Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walaupun tanpa <i>use case</i> tambahan itu.
5	Include		Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini.

(Yuni Sugiarti, 2013)

c. *Activity diagram*




Activity diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem. (Yuni Sugiarti, 2013).

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



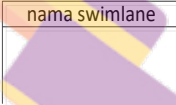


- 1) Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.
- 2) Urutan atau pengelompokan tampilan dari sistem / *user interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan.
- 3) Rancangan pengujian 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:

Table 2.4 Simbol *Activity Diagram*

No	Nama	Simbol	Keterangan
1	Status awal		Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
2	Aktivitas		Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3	<i>Decision</i>		Pilihan untuk pengambilan keputusan.

Tabel 2.4 Simbol *Activity Diagram* (Lanjutan)

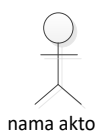

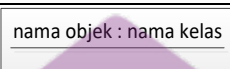
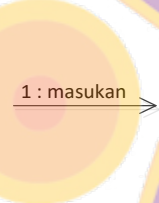
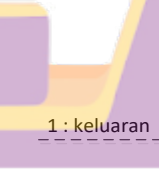
No	Nama	Simbol	Keterangan
4	<i>join</i>		Untuk menunjukkan kegiatan yang dilakukan secara paralel.
5	Status akhir		Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
6	Swimlane		Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.
7	Percabangan		Menunjukkan adanya dekomposisi.
8	Penggabungan		Tanda penerima.

d. *Sequence Diagram*

Sequence diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansi menjadi objek itu. (Yuni Sugiarti, 2013).

Berikut adalah simbol-simbol yang ada pada diagram *Sequence* :

Table 2.5 Simbol *Sequence Diagram*

No	Nama	Simbol	Keterangan
1	Aktor		Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri.
2	Garis hidup/ <i>lifeline</i>		Menyatakan kehidupan suatu objek
3	Objek		Menyatakan objek yang berinteraksi pesan
4	Waktu aktif		Menyatakan objek dalam keadaan aktif dan berinteraksi pesan
5	Pesan tipe <i>send</i>		Menyatakan bahwa suatu objek mengirimkan data / masukan / informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.
6	Pesan tipe <i>return</i>		Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian.

(Yuni Sugiarti,2013)

10. Perangkat Lunak Yang Digunakan

a. XAMPP

Menurut Puspitasari (2011), XAMPP adalah sebuah *software web server* apache yang didalamnya sudah tersedia *database server* mysql dan support php programming. xampp merupakan software yang mudah digunakan gratis dan mendukung instalasi di linux dan windows.

XAMPP juga merupakan paket PHP dalam MySQL berbasis *open source*, yang dapat digunakan sebagai *tool* pembantu pengembang aplikasi berbasis PHP, XAMPP mengombinasikan beberapa paket perangkat lunak berbeda ke dalam satu paket (Riyanto, 2011).

b. DBMS

DBMS merupakan perangkat lunak yang dirancang dapat melakukan utilisasi dan mengelola koleksi data dalam jumlah besar. DBMS juga dirancang untuk melakukan manipulasi data secara lebih mudah. DBMS merupakan antar muka pengguna basis data (baik pengguna langsung maupun aplikasi) dengan data yang tersimpan. Penyimpanan data oleh DBMS disesuaikan dengan bentuk model datanya, beberapa contoh DBMS adalah PostgreSQL, MySQL, DB2, Oracle, SQL Server, dan lain-lain (Utami dan Hartono, 2012).

Menurut Kadir (2014), DBMS (*database management system*) adalah program yang di tunjukan untuk melaksanakan manajemen data. Perangkat lunak ini menyediakan fasilitas untuk menyimpan data, memanipulasi data dan mengambil data dengan cara yang mudah dan cepat. Di lingkungan PC yang berbasis Windows, Microsoft Access adalah contoh DBMS yang sangat populer. Dilingkungan Linux, MySQL merupakan DBMS yang sangat banyak dipakai untuk aplikasi web.

c. *MySQL*

MySQL merupakan database yang pertama kali didukung oleh bahasa pemrograman script untuk internet (PHP dan Perl). *MySQL* dan PHP dianggap sebagai pasangan software pembangun aplikasi web yang ideal. *MySQL* lebih sering sering digunakan untuk membangun aplikasi berbasis web, umumnya pengembangan aplikasinya menggunakan bahasa pemrograman script PHP (Arief, M.Rudianto, 2011).

Menurut Madcoms (2011) *MySQL* adalah salah satu program yang dapat digunakan sebagai *database*, dan merupakan salah satu *software* untuk *database server* yang banyak digunakan. *MySQL* bersifat *Open Source* dan menggunakan *SQL*. *MySQL* bisa dijalankan diberbagai *platform* misalnya *Windows*, *Linux* dan lain sebagainya. *MySQL* memiliki beberapa kelebihan, antara lain.

- 1) *MySQL* dapat digunakan oleh beberapa user dalam waktu yang bersamaan tanpa mengalami masalah.
- 2) *MySQL* memiliki kecepatan yang bagus dalam menangani *query* sederhana.
- 3) *MySQL* memiliki operator dan fungsi secara fungsi secara penuh dan mendukung perintah *Select* dan *Where* dalam perintah *query*.
- 4) *MySQL* memiliki keamanan yang bagus karena beberapa lapisan sekuritas seperti level subnetmaks, nama host, dan izin akses user dengan system perijinan yang mendetail serta sandi terenskripsi.

- 5) *MySQL* mampu menangani basis data dalam jumlah skala besar, dengan jumlah rekaman (*record*) lebih dari 50 juta dan 60 ribu tabel serta kurang lebih 5 milyar baris. Selain itu batas indeks yang dapat ditampung mencapai 32 indeks pada tiap tabelnya.
- 6) *MySQL* dapat melakukan koneksi dengan *clien* dengan menggunakan protocol *TCP/IP*, *Unix socket* (UNIX), atau *Named Pipes* (NT).
- 7) *MySQL* dapat mendeteksi pesan kesalahan pada client dengan menggunakan lebih dari dua puluh bahasa.
- 8) *MySQL* dapat berjalan stabil pada berbagai sistem operasi seperti *Windows*, *Linux*, *FreeBSD*, *Mac Os X Server*, *Solaris*, *Amiga*, dan masih banyak lagi.
- 9) *MySQL* didistribusikan secara *Open source*, dibawah lisensi *GPL* sehingga dapat digunakan gratis.

11. Bahasa Pemrograman

a. *Hypertext Preprocessor* (PHP)

Menurut Suhartanto, (2012) singkatan dari PHP *Hypertext Preprocessor* yang digunakan sebagai bahasa script server_side dalam pengembangan Web yang disisipkan pada dokumen HTML. Penggunaan PHP memungkinkan Web dapat dibuat dinamis sehingga maintenance situs web tersebut menjadi lebih mudah dan efisien. PHP merupakan *software open-source* yang disebar dan dilisensikan secara gratis serta dapat didownload secara bebas.

b. *Hypertext Markup Language (HTML)*

Hypertext Markup Language (HTML) adalah bahasa *standard* yang digunakan untuk menampilkan halaman *web* (Hidayatullah dan Kawistara, 2017).

HTML (*Hyper Text Markup Language*). Dokumen HTML merupakan teks murni yang sering disebut dengan *web page*. Dokumen HTML berekstensi “.htm” atau “.html” (Madscom, 2011).

c. *Cascading Style Sheet (CSS)*

CSS adalah suatu cara untuk membuat format atau layout halaman web menjadi lebih menarik dan mudah dikelola (Husni, 2007).

Cascading Style Sheet (CSS) merupakan aturan untuk mengendalikan beberapa komponen dalam sebuah web sehingga akan lebih terstruktur dan seragam. CSS bukan merupakan bahasa pemrograman. CSS dapat mengendalikan ukuran gambar, warna bagian tubuh pada teks, warna tabel, ukuran *border*, warna *border*, warna *hyperlink*, warna *mouse over*, spasi antar paragraph, spasi antar teks, margin kiri, kanan, atas, bawah dan parameter lainnya. CSS adalah bahasa *style sheet* yang digunakan untuk mengatur tampilan dokumen (Sianipar, 2015).

B. Penelitian Sebelumnya

Beberapa penelitian sebelumnya yang timbulnya gagasan dan permasalahan yang akan diteliti untuk menguraikan teori, temuan, dan bahan

penelitian lain yang diperoleh dari acuan untuk dijadikan landasan dalam pelaksanaan penelitian ini adalah sebagai berikut:

1. Penelitian yang dilakukan oleh Pami (2017), yang berjudul “Sistem Pendukung Keputusan Pemilihan Karyawan Terbaik dengan Metode Promethee” bertujuan untuk menghasilkan aplikasi untuk menentukan karyawan terbaik sesuai dengan kriteria-kriteria yang telah ditentukan. Dalam penelitian ini peneliti menggunakan metode *PROMETHEE (Preference Ranking Organization Method for Enrichment Evaluations)*. Hasil dari penelitian ini memudahkan pihak manajemen perusahaan dalam pencarian data karyawan yang ingin dipromosikan untuk mendapatkan kenaikan pangkat, subsidi Haji ataupun balas jasanya dinaikkan.

Perbedaan yang dilakukan oleh Pami dengan penelitian yang dilakukan oleh penulis yaitu:

- a) Sistem pendukung keputusan yang dibuat oleh Pami menggunakan 8 kriteia yang berbeda dan penulis menggunakan 10 kriteria yang berbeda.
 - b) Sistem pendukung keputusan yang dibuat oleh Pami berbasis desktop sedangkan yang dibuat penulis berbasis website sehingga user tidak perlu menginstal aplikasi dan dapat mengakses di perangkat manapun.
2. Penelitian yang dilakukan oleh Wafi.dkk (2017), yang berjudul “Implementasi Metode Promethee II untuk Menentukan Pemenang Tender Proyek (Studi Kasus: Dinas Perhubungan dan LLAJ Provinsi Jawa Timur)” bertujuan untuk memudahkan pihak penyelenggara tender untuk menentukan pemenang tender dengan proses yang jelas dan mengurangi

praktik KKN dalam penyelenggaraan tender. Dalam penelitian ini peneliti menggunakan metode *Preference Ranking Organization Method for Enrichment Evaluations (PROMETHEE)*. Hasil dari penelitian ini adalah akurasi sistem penentuan pemilihan pemenang tender menggunakan metode PROMETHEE II memiliki tingkat akurasi tertinggi pada penggunaan tipe preferensi usual criterion dan quansi criterion yaitu sebesar 84.210%, sedangkan pada penggunaan tipe preferensi level criterion mencapai nilai terendah sebesar 63.157%.

Perbedaan yang dilakukan oleh Wafi.dkk dengan penelitian yang dilakukan oleh penulis yaitu:

Sistem pendukung keputusan yang dibuat oleh Wafi.dkk menggunakan bobot berbentuk persentase angka sedangkan penulis menggunakan angka desimal

3. Penelitian yang dilakukan oleh Safrizal.dkk (2016), yang berjudul “Multi-Criteria Decision Making Dalam Penentuan Jurusan Siswa Pada Lembaga Pendidikan dan Pelatihan (LPP) Penerbangan” bertujuan untuk membantu guru mendapatkan keputusan yang tepat dan optimal dalam menentukan jurusan yang cocok bagi siswa yang biasa dilakukan pada akhir semester 2 kelas X. . Dalam penelitian ini peneliti menggunakan metode *Analytical Hierarchy Process (AHP)*. Hasil dari penelitian ini dapat memudahkan pihak manajemen didalam menempatkan lulusan dalam On The Job Training (OJT) dalam proses penentuan jurusan siswa LPP Penerbangan.

Perbedaan yang dilakukan oleh Safrizal.dkk dengan penelitian yang dilakukan oleh penulis yaitu:

- a) Sistem pendukung keputusan yang dibuat oleh Safrizal.dkk menggunakan metode *Analytical Hierarchy Process* (AHP) sedangkan yang dibuat penulis menggunakan metode *Preference Ranking Organization Method for Enrichment Evaluations* (PROMETHEE).
 - b) Sistem pendukung keputusan yang dibuat oleh Safrizal.dkk berbasis desktop sedangkan yang dibuat penulis berbasis website sehingga user tidak perlu menginstal aplikasi dan dapat mengakses di perangkat manapun.
4. Penelitian yang dilakukan oleh Adila.dkk (2018), yang berjudul “sistem pendukung keputusan pemilihan tanaman pangan pada suatu lahan berdasarkan kondisi tanah dengan metode promethee” bertujuan untuk memudahkan petani dalam memilih bibit tanaman yang akan ditanam di lahan tertentu. Dalam penelitian ini peneliti menggunakan metode *Preference Ranking Organization Method for Enrichment Evaluations* (PROMETHEE). Hasil dari penelitian ini adalah Sistem mampu memberikan fasilitas– fasilitas penunjang yang dapat memberikan kemudahan bagi pengguna dalam mengambil informasi yang diberikan oleh sistem.

Perbedaan yang dilakukan oleh Adila.dkk dengan yang dilakukan oleh penulis yaitu:

Sistem pendukung keputusan yang dibuat oleh Adila.dkk menggunakan 12 kriteria sedangkan yang dibuat penulis 10 kriteria .



Tabel 2.6 Perbandingan Penelitian Terdahulu

No	Nama Peneliti (Tahun)	Judul	Hasil	Penelitian Sekarang
1	Setya Pami (2017).	Sistem Pendukung Keputusan Pemilihan Karyawan dengan Metode Promethee	Memudahkan pihak manajemen perusahaan dalam pencarian data karyawan yang ingin dipromosikan untuk mendapatkan kenaikan pangkat, subsidi Haji ataupun balas jasanya dinaikkan.	<ul style="list-style-type: none"> - Sistem pendukung keputusan yang dibuat oleh penulis menggunakan 10 kriteria yang berbeda. - Sistem pendukung keputusan yang dibuat oleh penulis berbasis website sehingga user tidak perlu menginstal aplikasi dan dapat mengakses di perangkat manapun.
2	Wafi.dkk (2017)	Implementasi Metode Promethee untuk Menentukan Pemenang Tender Proyek (Studi Kasus: Dinas Perhubungan dan LLAJ Provinsi Jawa Timur)	Akurasi sistem penentuan pemilihan pemenang tender menggunakan metode PROMETHEE memiliki tingkat akurasi tertinggi pada penggunaan tipe preferensi usual criterion dan quansi criterion yaitu sebesar 84.210%, sedangkan pada penggunaan tipe preferensi level criterion mencapai nilai terendah sebesar 63.157%.	Penelitian yang dilakukan wafi.dkk menggunakan bobot berbentuk angka persentase sedangkan penulis menggunakan angka desimal

Tabel 2.6 Perbandingan Penelitian Terdahulu (Lanjutan)

No	Nama Peneliti (Tahun)	Judul	Hasil	Penelitian Sekarang
3	Safrizal.dkk (2016)	Multi-Criteria Decision Making Dalam Penentuan Jurusan Siswa Pada Lembaga Pendidikan dan Pelatihan (LPP) Penerbangan	Memudahkan pihak manajemen didalam menempatkan lulusan dalam On The Job Training (OJT) dalam proses penentuan jurusan siswa LPP Penerbangan	<ul style="list-style-type: none"> - Menggunakan metode <i>Preference Ranking Organization Method for Enrichment Evaluations (PROMETHEE)</i>. - Sistem pendukung keputusan yang dibuat oleh penulis berbasis website sehingga user tidak perlu menginstal aplikasi dan dapat mengakses di perangkat manapun.
4	Adila.dkk (2018)	Sistem pendukung keputusan pemilihan tanaman pangan pada suatu lahan berdsarkan kondisi tanah dengan metode promethee	Sistem mampu memberikan fasilitas—fasilitas penunjang yang dapat memberikan kemudahan bagi pengguna dalam mengambil informasi yang diberikan oleh sistem.	Sistem pendukung keputusan yang dibuat oleh penulis menggunakan 10 kriteria