

## BAB II

### TINJAUAN PUSTAKA

Menurut Jurnal (Hilyah Magdalena;2012:191) Sistem Pendukung Keputusan *Decision Support System* (DSS) pada tahun 1970-an sebagai pengganti istilah *Management Information System*. Tetapi pada dasarnya Sistem Pendukung Keputusan merupakan pengembangan lebih lanjut dari *Management Information System* yang dirancang sedemikian rupa sehingga bersifat interaktif dengan pemakainya. Maksud dan tujuan dari adanya sistem pendukung keputusan, yaitu untuk mendukung pengambil keputusan memilih alternatif keputusan yang merupakan hasil pengolahan informasi-informasi yang diperoleh dengan menggunakan model-model pengambilan keputusan serta untuk menyelesaikan masalah yang bersifat terstruktur, semi struktur dan tidak terstruktur.

Pada dasarnya pengambilan keputusan adalah suatu pendekatan sistematis pada suatu masalah, pengumpulan fakta dan informasi, penentuan yang baik untuk alternatif yang dihadapi, dan pengambilan tindakan yang menurut analisis. Untuk kepentingan itu, sebagian besar pembuat keputusan dengan mempertimbangkan rasio manfaat/biaya, dihadapkan pada suatu keharusan untuk mengandalkan sistem yang mampu memecahkan suatu masalah secara efektif, yang kemudian disebut dengan sistem pendukung keputusan (SPK).

### **II.1.1 Pengertian Sistem**

Sistem merupakan kumpulan dari unsur atau elemen-elemen yang saling berkaitan/berinteraksi dan saling mempengaruhi dalam melakukan kegiatan bersama untuk mencapai suatu tujuan tertentu. Suatu sistem didefinisikan sebagai suatu kesatuan yang terdiri dari dua atau lebih komponen atau subsistem yang berintegrasi untuk mencapai suatu tujuan. Suatu system mempunyai karakteristik yaitu mempunyai komponen, batas sistem, lingkungan luar sistem, penghubung, masukan, keluaran, pengolah dan sasaran (Asbon Hendra;2011:157).

### **II.1.2 Pengertian Keputusan**

Keputusan merupakan kegiatan memilih suatu strategi atau tindakan dalam pemecahan masalah tersebut. Tindakan memilih strategi aksi yang diyakini akan memberikan solusi terbaik atas sesuatu itu disebut pengambilan keputusan. Keputusan merupakan sebuah kesimpulan yang dicapai sesudah dilakukan pertimbangan (analisa), yang terjadi setelah satu kemungkinan dipilih, sementara yang lain dikesampingkan.

### **II.1.3 Konsep Sistem Pendukung Keputusan**

Menurut Jurnal (Sri Winiarti;2013:329) sistem pendukung keputusan menurut *Gorry Dan Scout Morton* adalah sistem berbasis komputer interaktif, yang membantu para pengambil keputusan untuk menggunakan data dan berbagai model untuk memecahkan masalah-masalah tidak terstruktur.

Hal yang perlu diperhatikan di sini adalah bahwa keberadaan SPK bukan untuk menggantikan tugas manajer, tetapi untuk menjadi sarana penunjang bagi mereka. SPK merupakan implementasi teori-teori pengambilan keputusan yang telah diperkenalkan oleh ilmu-ilmu seperti *operation research dan management science*. Hanya bedanya adalah bahwa dahulu untuk mencari penyelesaian masalah yang dihadapi harus dilakukan perhitungan iterasi secara manual. Dalam kedua bidang ilmu di atas, dikenal istilah *decision modeling, decision theory, decision analysis* yang pada hakekatnya adalah merepresentasikan permasalahan manajemen yang dihadapi setiap hari ke dalam bentuk kuantitatif.

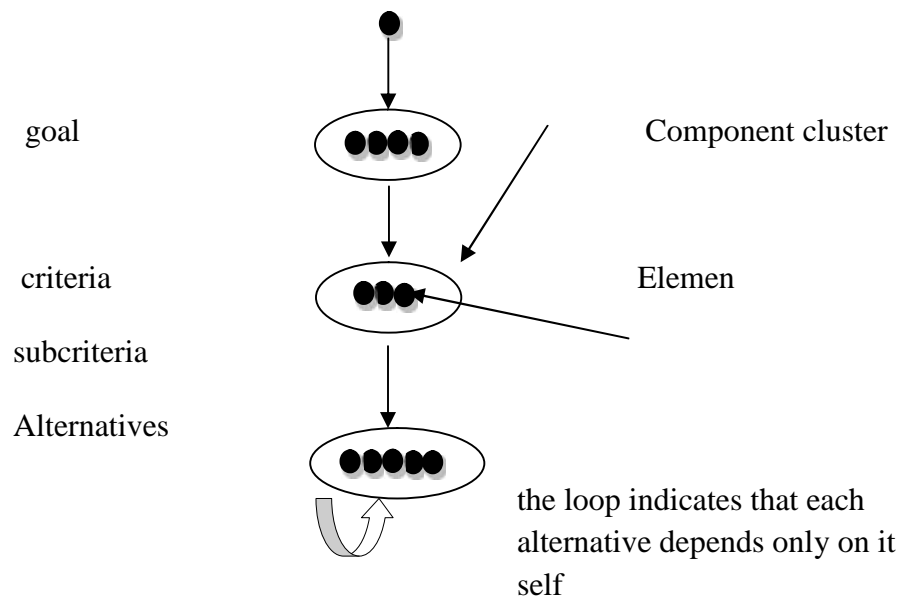
## **II.2. Analytical Hierarkhi Process (AHP)**

*Analytical Hierarkhi Process* merupakan suatu model pendukung keputusan yang dikembangkan oleh Thomas L. Saati. Model pendukung keputusan ini akan menguraikan masalah multi faktor atau multi kriteria yang kompleks menjadi suatu hirarki. Hirarki didefinisikan sebagai suatu representasi dari sebuah permasalahan yang kompleks dalam suatu struktur multi level dimana level pertama adalah tujuan, yang diikuti level faktor, kriteria, sub kriteria, dan seterusnya ke bawah hingga level terakhir dari alternatif. Dengan hirarki, suatu masalah yang kompleks dapat diuraikan ke dalam kelompok-kelompoknya yang kemudian diatur menjadi suatu bentuk hirarki sehingga permasalahan akan tampak lebih terstruktur dan sistematis (Hilyah Magdalena;2010:193). AHP sering digunakan sebagai metode pemecahan masalah disbanding dengan metode yang lain karena alasan-alasan sebagai berikut:

- a. Struktur yang berhirarki, sebagai konsekuensi dari kriteria yang dipilih, sampai pada subkriteria yang paling dalam.
- b. Memperhitungkan validitas sampai dengan batas toleransi inkonsistensi berbagai kriteria dan alternatif yang dipilih oleh pengambil keputusan.
- c. Memperhitungkan daya tahan output analisis sensitivitas pengambilan keputusan

AHP dikembangkan pada tahun 1970an oleh Dr Thomas L. Satty untuk menyediakan pendekatan sistematis untuk menentukan prioritas dan pengambilan keputusan dalam suatu kompleks lingkungan. AHP dirancang untuk mencerminkan cara berpikir orang sebenarnya. Metode ini memungkinkan aspek kuantitatif dan kualitatif keputusan yang akan dipertimbangkan. AHP mengurangi keputusan yang kompleks menjadi sebuah rangkaian satu-satu pada perbandingan yang kemudian memberikan hasil yang akurat. AHP juga menggunakan skala rasio untuk bobot kriteria dan scoring alternatif yang menambahkan untuk pengukuran presisi. Karena sulitnya menentukan bobot-bobot ataupun prioritas-prioritas yang sering berubah-ubah, digunakan perbandingan berpasangan yang menggunakan data, pengetahuan, dan pengalaman untuk memperoleh prioritas.

Berikut gambar Fase Proses Pengambilan Keputusan struktur *Analytical Hierarchy Process* (AHP) :



**Gambar II.I. Fase Proses Pengambilan Keputusan Struktur AHP**

Sumber : Jurnal Hilyah Magdalena ; 2012 : 193

### II.2.1 Landasan Aksiomatik *Analytical Hierarchy Process* ( AHP)

*Analytic Hierarchy Process* (AHP) mempunyai landasan aksiomatik yang terdiri dari :

1. *Reciprocal Comparison*, yang mengandung arti si pengambil keputusan harus bisa membuat perbandingan dan menyatakan preferensinya. Preferensinya itu sendiri harus memenuhi syarat resiprokal yaitu kalau A lebih disukai dari B dengan skala  $x$ , maka B lebih disukai dari A dengan skala  $\frac{1}{x}$ .
2. *Homogeneity*, yang mengandung arti preferensi seseorang harus dapat dinyatakan dalam skala terbatas atau dengan kata lain elemen-elemennya dapat dibandingkan satu sama lain. Kalau aksioma ini tidak dapat dipenuhi maka

elemen-elemen yang dibandingkan tersebut tidak homogenous dan harus dibentuk suatu 'cluster' (kelompok elemen-elemen) yang baru.

3. *Independence*, yang berarti preferensi dinyatakan dengan mengasumsikan bahwa kriteria tidak dipengaruhi oleh alternatif-alternatif yang ada melainkan oleh objektif secara keseluruhan. Ini menunjukkan bahwa pola ketergantungan atau pengaruh dalam model AHP adalah searah keatas, Artinya perbandingan antara elemen-elemen dalam satu level dipengaruhi atau tergantung oleh elemen-elemen dalam level di atasnya.
4. *Expectations*, artinya untuk tujuan pengambilan keputusan, struktur hirarki diasumsikan lengkap. Apabila asumsi ini tidak dipenuhi maka si pengambil keputusan tidak memakai seluruh kriteria dan atau objektif yang tersedia atau diperlukan sehingga keputusan yang diambil dianggap tidak lengkap (Hilyah Magdalena;2012:194.)

## **II.2.2. Tahapan – Tahapan Proses Metode *Analytic Hierarchy Process* (AHP)**

Menurut Jurnal (Yusuf Anshori;2012:127) menjelaskan bahwa secara umum, tahapan-tahapan proses yang harus dilakukan dalam menggunakan AHP untuk memecahkan suatu masalah adalah sebagai berikut :

1. Mendefenisikan permasalahan dan menentukan tujuan. Bila AHP digunakan untuk memilih alternatif atau menyusun prioritas alternatif, maka tahap ini dilakukan pengembangan alternatif.

2. Menyusun masalah ke dalam suatu struktur hierarki sehingga permasalahan yang kompleks dapat ditinjau dari sisi yang detail dan terukur .
3. Menyusun prioritas untuk tiap elemen masalah pada setiap hierarki. Prioritas ini dihasilkan dari suatu matriks perbandingan berpasangan antara seluruh elemen pada tingkat hierarki yang sama.
4. Melakukan pengujian konsistensi terhadap perbandingan antar elemen yang didapatkan pada tiap tingkat hierarki. Thomas L. Saaty membuktikan bahwa Indeks Konsistensi dari matriks berordo  $n$  diperoleh rumus sebagai berikut:

$$CI = \frac{\lambda_{max} - n}{n - 1} \dots\dots\dots \text{Pers}(2.1)$$

Dimana :

$CI$  = *Consistency Index* ( Rasio penyimpangan konsistensi )

$\lambda_{max}$  = Nilai eigen terbesar dari matriks berordo  $n$

$n$  = jumlah elemen yang dibandingkan

Nilai  $CI$  bernilai nol apabila terdapat standar untuk menyatakan apakah  $CI$  menunjukkan matriks yang konsisten. Saaty berpendapat bahwa suatu matriks yang dihasilkan dari perbandingan yang dilakukan secara acak merupakan suatu matriks yang tidak konsisten. Dari matriks acak didapatkan juga nilai *Consistency Index* yang disebut dengan *Random Index (RI)*.

Dengan membandingkan  $CI$  dengan  $RI$  maka didapatkan patokan untuk menentukan tingkat konsistensi suatu matriks yang disebut dengan *Consistency Ratio (CR)* dengan rumus :

$$CR = CI / RI$$

.....Pers(2.2)

Dimana :

$CR$  = Consistency Ratio

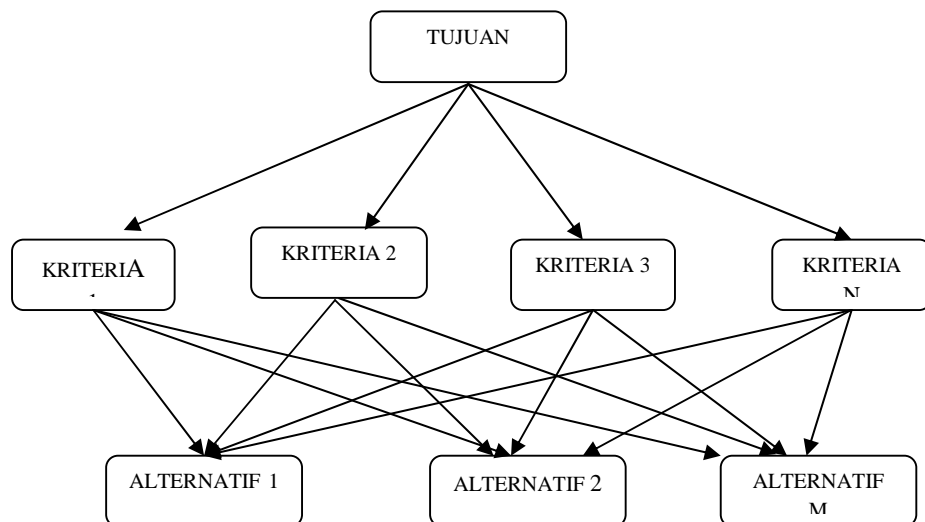
$RI$  = *Random Index*

### II.2.3. Prinsip Dasar *Analytic Hierarchy Process* ( **AHP** )

Metode AHP ada beberapa prinsip dasar yang harus diperhatikan yaitu :

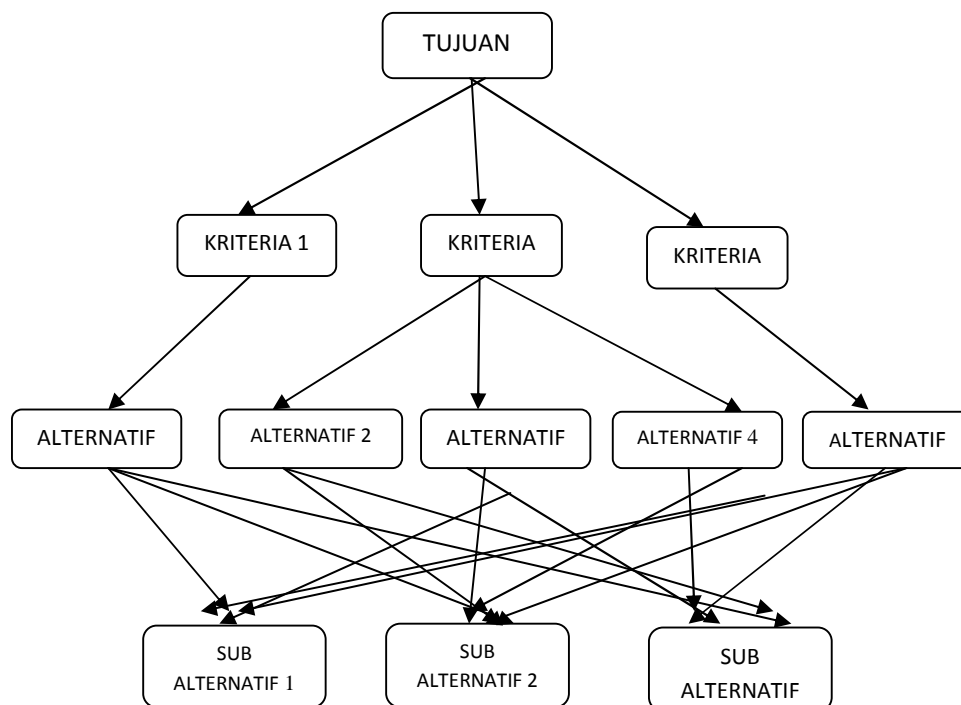
#### 1. Membuat Hirarki

Sistem yang kompleks bisa dipahami dengan memecahnya menjadi elemen elemen pendukung, menyusun elemen secara hirarki.



**Gambar II.2 Struktur Hirarki Complete**

Sumber : Jurnal Yusuf Anshori, Struktur Hirarki *Complete* ; 2012 : 128



**Gambar II.3. Struktur Hirarki Incomplete**

Sumber : Jurnal Yusuf Anshori, Struktur Hirarki *Incomplete* ; 2012 : 128

Pada Gambar II.2 Tingkat pertama merupakan tujuan keputusan (goal), tingkat kedua : kriteria-kriteria, tingkat ketiga : alternatif-alternatif , Hirarki masalah disusun untuk membantu proses pengambilan keputusan Dengan memperhatikan seluruh elemen keputusan yang terlibat dalam sistem. Sebagian besar masalah menjadi sulit untuk diselesaikan karena proses pemecahannya dilakukan tanpa memandang masalah sebagai suatu sistem dengan suatu struktur tertentu. Sedangkan pada Gambar II.3 hirarki *Incomplete* sama dengan hirarki *complete* hanya saja di *Incomplete* ada tingkat keempat dari alternatif menjadi sub alternatif, Kriteria dan alternatif dilakukan dengan perbandingan berpasangan.

## 2. *Synthesis of Priority*

*Synthesis of Priority* dilakukan dengan menggunakan eigen vector method untuk mendapatkan bobot relatif bagi unsur – unsur pengambilan keputusan.

## 3. *Logical Consistency*

*Logical Consistency* merupakan karakteristik penting AHP. Hal ini dicapai dengan mengagresikan seluruh eigen vector yang diperoleh dari berbagai tingkatan hirarki dan selanjutnya diperoleh suatu vector compositive tertimbang yang menghasilkan urutan pengambilan keputusan.

**Tabel II.1. Matriks Perbandingan Berpasangan**

	$A_1$	$A_2$		$A_n$
$A_1$	$a_{11}$	$a_{12}$		$a_{1n}$
$A_2$	$a_{21}$	$a_{22}$		$a_{2n}$
$A_n$	$a_{n1}$	$a_{n2}$		$a_{nn}$

Sumber : Jurnal Yusuf Anshori, Matriks Perbandingan ; 2012 : 128

Berdasarkan tabel di atas kita dapat menentukan skala perbandingan antar elemen dalam proses pengambilan keputusan. Nilai  $B_{11}$  adalah nilai perbandingan elemen  $B_1$  ( baris) terhadap  $B_1$  ( kolom) yang menyatakan hubungan :

- a. Seberapa jauh tingkat kepentingan  $B_1$  (baris) terhadap kriteria dibandingkan dengan  $B_1$  (kolom) .

- b. Seberapa jauh dominal  $B_1$  (baris) terhadap  $B_1$  (kolom) .
- c. Seberapa banyak sifat kriteria terhadap pada  $B_1$  (baris) dibandingkan dengan  $B_1$  (kolom).

Nilai numerik yang dikenakan untuk seluruh perbandingan diperoleh dari skala perbandingan 1 sampai 9 yang telah ditetapkan oleh Saaty, seperti pada table II.2.

**Tabel II.2. Penilaian Perbandingan Berpasangan**

<b>Tingkat Kepentingan</b>	<b>Definisi</b>
1	Sama pentingnya dibanding yang lain
3	Modern ( cukup ) pentingnya dibanding yang lain
5	Kuat pentingnya dibanding yang lain
7	Sangat kuat pentingnya dibanding yang lain
9	Ekstrim pentingnya dibanding yang lain
2,4,6,8	Nilai dimana dua nilai yang berdekatan
Resiprokal	Jika elemen i memiliki salah satu angka diatas ketika dibandingkan elemen j, maka j memiliki kebalikannya ketika disbanding elemen i

Sumber : Jurnal Yusuf Anshori, Matriks Berpasangan ; 2012 : 128

Model AHP didasarkan pada *pair – wise comparison matrix*, di mana elemen elemen pada matriks tersebut merupakan *judgement* dari *decision maker*. Seorang *decision maker* akan memberikan penilaian, mempersepsikan, atau memperkirakan kemungkinan dari suatu hal yang dihadapi. Matriks tersebut terdapat pada setiap level of cluster dari suatu struktur model AHP yang membagi habis suatu persoalan.

Berikut ini contoh suatu *Pair – Wise Comparison Matrix* pada suatu *level of cluster* yaitu :

**Tabel II.3. Tabel *Pair – Wise Comparison Matrix***

Kriteria	W	X	Y	Z
W	1/1	5/1	6/1	9/1
X	1/5	1/1	1/5	1/6
Y	1/6	5/1	1/1	4/1
Z	1/9	6/1	1/4	1/1

Sumber : Jurnal Yusuf Anshori, *Wise Comparison Matrix* ; 2012 : 128

Baris 1 kolom 2 : Jika W dibandingkan dengan X, maka W lebih penting /disukai/dimungkinkan daripada X yaitu sebesar 5, artinya : *W essential or strong importance* daripada X, dan seterusnya. Angka 5 bukan berarti bahwa W lima kali lebih besar dari X, tetapi W *strong importance* dibandingkan dengan X. Dengan demikian pada baris 1 kolom 2 diisi dengan kebalikan dari 5 yakni 1/5. Jika W dibandingkan dengan Z, maka W *extreme importance* daripada Z dengan nilai *judgement* sebesar 9. Jadi baris 1 kolom 4 diisi dengan nilai 9 dan seterusnya.

**Tabel II.4. Nilai Random Index**

<b>N</b>	<b>Random Index</b>
1	0,000
2	0,000
3	0,580
4	0,900
5	1,120
6	1,240
7	1,320
8	1.410
9	1,450
10	1,490
11	1,510
12	1,480
13	1.560
14	1,570
15	1,590

Sumber : Jurnal Yusuf Anshori, *Random Indexs* ; 2012 : 129

### **II.3. UML ( *Unifed Modeling Language* )**

Menurut Windu Gata (2013:4) Hasil pemodelan pada OOAD terdokumentasikan dalam bentuk *Unified Modeling Language* (UML). UML

adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak.

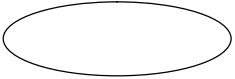
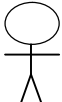
UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem. UML saat ini sangat banyak dipergunakan dalam dunia industri yang merupakan standar bahasa pemodelan umum dalam industri perangkat lunak dan pengembangan sistem.




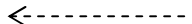
Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis UML adalah sebagai berikut :

#### 1. *Use case* Diagram

*Use case* diagram merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Dapat dikatakan *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Simbol-simbol yang digunakan dalam *use case* diagram, yaitu :

**Tabel II.5. Simbol *Use Case***

Gambar	Keterangan
	<i>Use case</i> menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal nama <i>use case</i> .
	Aktor adalah <i>abstraction</i> dari orang atau sistem yang lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasi aktor, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang




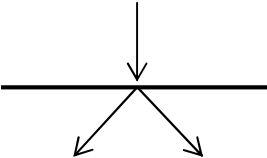
	berkaitan dengan peran pada konteks target sistem. Orang atau sistem bisa muncul dalam beberapa peran. Perlu dicatat bahwa aktor berinteraksi dengan <i>use case</i> , tetapi tidak memiliki control terhadap <i>use case</i> .
	Asosiasi antara aktor dan <i>use case</i> , digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengindikasikan aliran data.
	Asosiasi antara aktor dan <i>use case</i> yang menggunakan panah terbuka untuk mengindikasikan bila aktor berinteraksi secara pasif dengan sistem.
	<i>Include</i> , merupakan di dalam <i>use case</i> lain ( <i>required</i> ) atau pemanggilan <i>use case</i> oleh <i>use case</i> lain, contohnya adalah pemanggilan sebuah fungsi program.
	<i>Extend</i> , merupakan perluasan dari <i>use case</i> lain jika kondisi atau syarat terpenuhi.

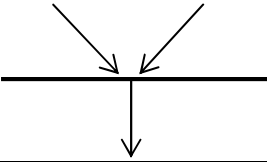
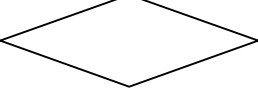

(Sumber : Windu Gata & Grace Gata; 2013 : 4)

## 2. Diagram Aktivitas (*Activity Diagram*)

*Activity Diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Simbol-simbol yang digunakan dalam *activity diagram*, yaitu :

**Tabel II.6. Simbol *Activity Diagram***

Gambar	Keterangan
	<i>Start point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktifitas.
	<i>End point</i> , akhir aktifitas.
	<i>Activites</i> , menggambarkan suatu proses/kegiatan bisnis.
	<i>Fork</i> (Percabangan), digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau untuk menggabungkan dua kegiatan pararel menjadi satu.

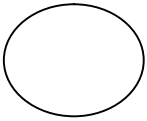
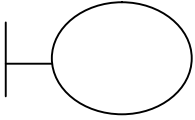
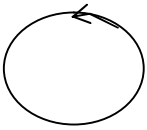
	<i>Join</i> (penggabungan) atau rake, digunakan untuk menunjukkan adanya dekomposisi.
	<i>Decision Points</i> , menggambarkan pilihan untuk pengambilan keputusan, <i>true</i> , <i>false</i> .
	<i>Swimlane</i> , pembagian <i>activity</i> diagram untuk menunjukkan siapa melakukan apa.

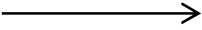
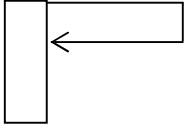


(Sumber : Windu Gata & Grace Gata ; 2013 : 6)

### 3. Diagram Urutan (*Sequence Diagram*)

*Sequence diagram* menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek. Simbol-simbol yang digunakan dalam *sequence diagram*, yaitu :

**Tabel II.7. Simbol *Sequence Diagram***

Gambar	Keterangan
	<i>Entity Class</i> , merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.
	<i>Boundary Class</i> , berisi kumpulan kelas yang menjadi <i>interface</i> atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan formentry dan <i>form</i> cetak.
	<i>Control class</i> , suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek.

	<p><i>Message</i>, simbol mengirim pesan antar <i>class</i>.</p>
	<p><i>Recursive</i>, menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.</p>
	<p><i>Activation</i>, <i>activation</i> mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi.</p>
	<p><i>Lifeline</i>, garis titik-titik yang terhubung dengan objek, sepanjang <i>lifeline</i> terdapat <i>activation</i>.</p>

(Sumber : Windu Gata & Grace Gata; 2013 : 7)

#### 4. *Class Diagram* (Diagram Kelas)

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.

*Class diagram* juga menunjukkan atribut-atribut dan operasi-operasi dari sebuah kelas dan *constraint* yang berhubungan dengan objek yang dikoneksikan. *Class diagram* secara khas meliputi: Kelas (*Class*), Relasi, *Associations*, *Generalization* dan *Aggregation*, Atribut (*Attributes*), Operasi (*Operations/Method*), *Visibility*, tingkat akses objek eksternal kepada suatu operasi atau atribut. Hubungan antar kelas mempunyai keterangan yang disebut dengan *multiplicity* atau kardinaliti.

**Tabel II.8. Multiplicity Class Diagram**

<b>Multiplicity</b>	<b>Penjelasan</b>
1	Satu dan hanya satu
0..*	Boleh tidak ada atau 1 atau lebih
1..*	1 atau lebih
0..1	Boleh tidak ada, maksimal 1
n..n	Batasan antara. Contoh 2..4 mempunyai arti minimal 2 maksimum 4

(Sumber : Windu Gata & Grace Gata ; 2013 : 8)

## II.4. MySQL

*MySQL* adalah database server relasional yang gratis di bawah lesensi GNU (*General Public License*). Dengan sifatnya *Open Source*, memungkinkan juga user untuk melakukan modifikasi pada *source code*-nya untuk memenuhi kebutuhan spesifik. *MySQL* merupakan database server *multi-user* dan *multi-threaded* yang tangguh (*robust*). Dengan memiliki banyak feature *MySQL* bisa bersaing dengan database komersil lainnya. Tidak mengejutkan, *MySQL* menjadi database pilihan untuk banyak pengguna PHP.

*MySQL* adalah salah jenis database server yang sangat terkenal. *MySQL* termasuk jenis RDMS (*Relation Database Manajement Sistem*). *MySQL* ini mendukung bahasa pemograman PHP. *MySQL* juga mempunyai query atau bahasa SQL (*Structured Query Language*) yang simple dan menggunakan *escape character* yang sama dengan PHP.(Ardhana, YM Kusuma;2012:88).

## II.5 Dreamweaver 8

*Dreamweaver* adalah suatu bentuk program editor web yang dibuat oleh macromeia dengan alamat *web site*. Dengan menggunakan program ini, seorang programmer web dapat dengan mudah membuat dan mendesain webnya. *Dreamweaver* sebagai editor yang komplet juga dapat digunakan untuk membuat animasi sederhana yang berbentuk *layer* dengan bantuan JavaScript yang didukungnya. Dengan adanya program ini kita tidak akan susah untuk mengetik skrip-skrip format HTML, PHP, JSP, ASP, JavaScript, CSS maupun brntuk program lainnya.

Sebagai editor, *Dreamweaver* mempunyai sifat yang WYSIWYG dibaca (wai-si-wing) yang artinya apa yang kita lihat pada halaman desain, maka semuanya itu akan kita peroleh pada browser. Dengan kelebihan ini sehingga programmer (pembuat program) dapat langsung melihat hasil buatannya tanpa harus membukanya pada aplikasi pengakses web seperti *Internet Explorer*, *Mozila*, dll). (Ardhana, YM Kusuma;2012:55).