

BAB II

LANDASAN TEORI

II.1. Sistem Pendukung Keputusan

Sistem pendukung keputusan (*Decision Support Systems* disingkat DSS) adalah bagian dari sistem informasi berbasis komputer (termasuk sistem berbasis pengetahuan (manajemen pengetahuan) yang dipakai untuk mendukung keputusan dalam suatu organisasi atau perusahaan. Dapat juga dikatakan sebagai sistem komputer yang mengolah data menjadi informasi untuk mengambil keputusan dari masalah semiterstruktur yang spesifik.

Sistem pendukung pengambilan keputusan kelompok (DSS) adalah sistem berbasis komputer yang interaktif, yang membantu pengambil keputusan dalam menggunakan data dan model untuk menyelesaikan masalah yang tidak terstruktur. Sistem pendukung ini membantu pengambilan keputusan manajemen dengan menggabungkan data, model-model dan alat-alat analisis yang kompleks, serta perangkat lunak yang akrab dengan tampilan pengguna ke dalam satu sistem yang memiliki kekuatan besar (*powerful*) yang dapat mendukung pengambilan keputusan yang semi atau tidak terstruktur. DSS menyajikan kepada pengguna satu perangkat alat yang fleksibel dan memiliki kemampuan tinggi untuk analisis data penting. Dengan kata lain, DSS menggabungkan sumber daya intelektual seorang individu dengan kemampuan komputer dalam rangka meningkatkan kualitas pengambilan keputusan. DSS diartikan sebagai tambahan bagi para

pengambil keputusan, untuk memperluas kapabilitas, namun tidak untuk menggantikan pertimbangan manajemen dalam pengambilan keputusannya.

Penyusunan sebuah model keputusan merupakan suatu cara untuk mengembangkan hubungan-hubungan logis yang mendasari persoalan keputusan kedalam suatu bentuk model matematis, yang mencerminkan hubungan di antara faktor-faktor yang terlibat. Proses pengambilan keputusan terdiri dari empat fase, yaitu :

1. Penelusuran (*Intelligence*)

Tahap ini merupakan tahap pendefinisian masalah serta identifikasi informasi yang dibutuhkan yang berkaitan dengan persoalan yang dihadapi serta keputusan yang akan diambil.

2. Perancangan (*Design*)

Tahap ini merupakan suatu proses untuk merepresentasikan model sistem yang akan dibangun berdasarkan pada asumsi yang telah ditetapkan. Dalam tahap ini, suatu model dari masalah dibuat, diuji, dan divalidasi.

3. Pemilihan (*Choice*)

Tahap ini merupakan suatu proses melakukan pengujian dan memilih keputusan terbaik berdasarkan kriteria tertentu yang telah ditentukan dan mengarah kepada tujuan yang akan dicapai.

4. Implementasi (*Implementation*)

Tahap ini merupakan tahap pelaksanaan dari keputusan yang telah diambil. Pada tahap ini perlu disusun serangkaian tindakan yang

terencana, sehingga hasil keputusan dapat dipantau dan disesuaikan apabila diperlukan perbaikanperbaikan.

II.1.1. Komponen – Komponen Sistem Pendukung Keputusan

Komponen - komponen sistem pendukung keputusan dapat diuraikan dalam beberapa subsistem sebagai berikut :

1. Subsistem Manajemen Database

Data Base Management System (DBMS) merupakan komponen penting dari suatu sistem pendukung keputusan, karena terdapat perbedaan kebutuhan data. *Database* merupakan mekanisme integrasi berbagai jenis data internal dan eksternal. Sebuah pengelolaan *database* yang efektif dapat menunjang segala aktivitas manajemen, terutama perannya sebagai fungsi utama penyajian informasi dalam pembuatan keputusan.

Kemampuan yang dibutuhkan dari manajemen *database* adalah sebagai berikut :

- a. Kemampuan untuk mengombinasikan berbagai data melalui pengambilan ekstaksi data.
- b. Kemampuan untuk menambahkan sumber data secara cepat dan mudah.
- c. Kemampuan untuk mengelola berbagai variasi data.

2. Subsistem Manajemen Basis Model

Salah satu keunggulan sistem pendukung keputusan adalah kemampuan untuk mengintegrasikan akses data dan model-model keputusan. Model cenderung tidak mencukupi karena adanya kesulitan dalam mengembangkan model yang terintegrasi untuk menangani sekumpulan keputusan yang saling bergantung. Cara untuk menangani persoalan ini dengan menggunakan koleksi berbagai model yang terpisah, dimana setiap model digunakan untuk menangani bagian yang berbeda dari masalah tersebut. Komunikasi antara berbagai model yang saling berhubungan diserahkan kepada pengambil keputusan sebagai proses intelektual dan manual.

3. Subsistem Perangkat Lunak Penyelenggara Dialog

Kekuatan dan fleksibilitas dari sistem pendukung keputusan timbul dari kemampuan interaksi antara sistem dan pemakai, yang dinamakan subsistem dialog. Bennet membagi subsistem dialog menjadi tiga bagian, yaitu :

- a. Bahasa aksi, meliputi apa yang dapat digunakan oleh pemakai dalam berkomunikasi dengan sistem. Hal ini meliputi pemilihan-pemilihan seperti papan ketik (*keyboard*), panel-panel sentuh, *joystick* perintah suara dan sebagainya.
- b. Bahasa tampilan dan presentasi, meliputi apa yang dapat digunakan untuk menampilkan sesuatu. Bahasa tampilan meliputi pilihan-pilihan

seperti printer, layar tampilan, grafik, warna, keluaran suara dan sebagainya.

- c. Basis pengetahuan, meliputi apa yang harus diketahui oleh pemakai agar pemakaian sistem bisa efektif. Basis pengetahuan dapat berada dalam pikiran pemakai, pada kartu referensi atau petunjuk, dalam buku manual dan sebagainya.

Kemampuan yang dimiliki sistem pendukung keputusan untuk mendukung dialog pemakai sistem meliputi :

- a. Kemampuan untuk menangani berbagai dialog, bahkan jika mungkin untuk mengombinasikan berbagai gaya dialog sesuai dengan pilihan pemakai.
- b. Kemampuan untuk mengakomodasikan tindakan pemakai dengan berbagai peralatan masukan.
- c. Kemampuan untuk menampilkan data dengan berbagai format dan peralatan keluaran.
- d. Kemampuan untuk memberikan dukungan yang fleksibel untuk mengetahui basis pengetahuan pemakai. (Mukhlida Fatmi Universitas Sumatra Utara : 2011)

II.2. Metode *Promethee*

Metode *Promethee* termasuk kedalam kelompok pemecahan masalah *Multi Criteria Decision Making (MCDM)* atau pengambilan keputusan kriteria majemuk yang merupakan disiplin ilmu yang sangat penting dalam pengambilan keputusan

atau suatu masalah yang memiliki lebih dari satu kriteria (multikriteria). Menurut Brans dan Marcchal, *Promethee* yang merupakan singkatan dari *Preference Ranking Organization Method for Enrichment Evaluation* adalah metode *outranking* yang menawarkan cara yang fleksibel dan sederhana kepada user (pembuat keputusan) untuk menganalisis masalah-masalah multikriteria. Prinsip yang digunakan adalah penetapan prioritas alternatif yang telah ditetapkan berdasarkan pertimbangan dengan kaidah dasar :

$$\text{Max } \{f_1(x), f_2(x), f_3(x), \dots, f_i(x), \dots, f_k(x)\} \text{ Ix.}$$

Dimana k adalah sejumlah kumpulan alternatif dan f ($i = 1, 2, \dots, k$) merupakan nilai/ukuran relatif kriteria untuk masing-masing alternatif. Termasuk dalam keluarga dari metode *outranking* yang dikembangkan oleh B.Roy dan meliputi dua fase :

1. Membangun hubungan *outranking* dari K , dimana K adalah sejumlah kumpulan alternaif.
2. 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 :

Tabel II.1. Data Dasar Analisis Promethee

No.	a_x	$f_1()$	$f_2()$	$f_j()$	$f_k()$
1.	A_1	$f_1(a_1)$	$f_2(a_1)$	$f_j(a_1)$	$f_k(a_1)$
2.	A_2	$f_1(a_2)$	$f_2(a_2)$	$f_j(a_2)$	$f_k(a_2)$
3.
4.	a_i	$f_1(a_i)$	$f_2(a_i)$	$f_j(a_i)$	$f_k(a_i)$
5.
6.	a_n	$f_1(a_n)$	$f_2(a_n)$	$f_j(a_n)$	$f_k(a_n)$

Sumber : Jurnal Dewi Safitri Hutabarat : 2013 ; 14)

Dimana :

A_1 : alternatif i

$f_k(a_i)$: kriteria yang ditetapkan untuk alternatif i struktural preferensi yang dibangun atas dasar kriteria.

$$\left. \begin{array}{l} \forall a, b \in A \quad f(a) > f(b) \\ F(a), f(b) \quad f(a) = f(b) \end{array} \right\} \begin{array}{l} a p b \\ n I b \end{array}$$

Struktur kriteria diatas mempunyai pengertian bahwa setiap alternatif a dan b yang merupakan elemen himpunan A, apabila nilai alternatif a unuk kriteria yang ditetapkan untuk alternatif a lebih dari nilai dari alternatif b, maka alternatif a lebih dipilih (*prefer*) dari pada alernatif b, sedangkan jika nilai dari alternatif sama dengan nilai dari alternatif b, maka dapat disimpulkan bahwa alternatif a tidak mempunyai perbedaan (*indifference*) dengan fungsi b, sehingga untuk menentukan alternatif mana yang lebih diprioritaskan dilakukan dengan memperhatikan nilai dari alternatif lainnya.

- a. Untuk mempermudah perhitungan dengan promethee, setiap alternatif menggunakan kode 1sampai jumlah alternatif untuk setiap jenis alternatif. Berikut ini adalah data alternatif siswa yang akan dipilih dapat dilihat tabel II.2.

Tabel II.2. Data Alternatif Siswa

No.	Nama Alternatif
1.	Budi = A
2.	Angga = B
3.	Morantika = C

Sumber : Jurnal Dewi Safitri Hutabarat : 2013 ; 15)

b. Menentukan Beberapa Kriteria

Kriteria yang digunakan didasarkan pada kebutuhan dalam proses pengambilan keputusan. Adapun kriterianya dapat dilihat pada tabel 3.

Tabel II.3. Data Kriteria

F1 (Pekerjaan Ayah)	PNS / Pegawai (bobot 1)
	Wiraswasta (bobot 2)
	Karyawan / Buruh (bobot 3)
	Guru Swasta (bobot 4)
F2 (Pekerjaan Ibu)	PNS / Pegawai (bobot 1)
	Wiraswasta (bobot 2)
	Karyawan/Buruh (bobot 3)
	Guru Swasta (bobot 4)
	Ibu Rumah Tangga (bobot 5)
F3 (Jumlah Penghasilan Orang Tua)	>= Rp.5000.000 (bobot 1)
	>= Rp.4000.000 (bobot 2)
	>= Rp.3000.000 (bobot 3)
	>= Rp.2000.000 (bobot 4)
	>= Rp.1000.000 (bobot 5)
F4 (Jumlah Tanggungan Orang Tua)	1 orang (bobot 1)
	2 orang (bobot 2)
	3 orang (bobot 3)
	4 orang (bobot 4)
	5 orang (bobot 5)
F5 (Kepemilikan Rumah)	Tidak layak (bobot 1)
	Kurang layak (bobot 2)
	Cukup layak (bobot 3)
	Layak (bobot 4)
	Sangat layak (bobot 5)

Sumber : Jurnal Dewi Safitri Hutabarat : 2013 ; 15)

Dari tabel alternatif dan tabel kriteria dibuatkan data dasar untuk evaluasi dengan metode *promethee*. Tapi terlebih dahulu sebelum data dimasukkan kedalam tabel untuk evaluasi dengan metode *promethee*, data alternatif dan kriteria dibuatkan simbolnya agar memudahkan dalam proses *promethee*.

1). Pemberian Simbol untuk Data Alternatif

Tabel II.4. Simbol Data Alternatif

Simbol	Nama Alternatif
1.	Budi = A
2.	Angga = B
3.	Morantika = C

Sumber : Jurnal Dewi Safitri Hutabarat : 2013 ; 15)

2). Pemberian Simbol Untuk Data Kriteria

Tabel II.5. Simbol Data Alternatif

Simbol	Nama Kriteria
F1	Pekerjaan Ayah
F2	Pekerjaan Ibu
F3	Penghasilan Orang Tua
F4	Jumlah Tanggungan Orang Tua
F5	Kepemilikan Rumah

Sumber : Jurnal Dewi Safitri Hutabarat : 2013 ; 15)

Tabel II.6. Contoh Kasus Nilai Kriteria Tiap Alternatif

Kriteria	Min Max	A	B	C	Tipe
F1(.)	Max	2	2	2	1
F2(.)	Max	3	2	2	1
F3(.)	Max	2	2	2	1
F4(.)	Max	4	5	4	1
F5(.)	Max	2	1	2	1

Sumber : Jurnal Dewi Safitri Hutabarat : 2013 ; 15)

Keterangan:

F1(.) = Pekerjaan Ayah

F2(.) = Pekerjaan Ibu

F3(.) = Penghasilan Orang Tua

F4(.) = Jumlah Tanggungan Orang Tua

F5(.) = Kepemilikan Rumah

A = Budi, di mana $f_1(A) = 2$, $f_2(A) = 3$, $f_3(A) = 2$, $f_4(A) = 4$, $f_5(A) = 2$.

B = Angga, di mana $f_1(B) = 2$, $f_2(B) = 2$, $f_3(B) = 2$, $f_4(B) = 3$, $f_5(B) = 1$.

C = Morantika, di mana $f_1(C) = 2$, $f_2(C) = 2$, $f_3(C) = 2$, $f_4(C) = 4$, $f_5(C) = 2$.

Tipe 1 = Kriteria Umum (Usual Criteria). (Jurnal Dewi Safitri Hutabarat : volume 1 nomor 1 ; 2013).

II.3. Beasiswa

Beasiswa adalah pemberian berupa bantuan keuangan yang di berikan kepada perorangan yang bertujuan untuk digunakan demi keberlangsungan pendidikan yang ditempuh. Pemberian beasiswa dapat di kategorikan pada pemberian cuma – cuma ataupun pemberian dengan ikatan kerja (biasa disebut ikatan dinas) setelah selesainya pendidikan. (Jurnal Apriansyah Putra : Vol 3 no 1 ; 2011)

Adapun kriteria kriteria yang menjadi dasar penerimaan beasiswa pada SMP Al – Ulum Medan yang harus diisi dalam formulir pendaftaran yaitu :

1. No NIS
2. Nama Siswa
3. Nama Orangtua
4. Pekerjaan Orangtua
5. Penghasilan
6. Jumlah Tanggungan
7. Kepemilikan Rumah

II.4. Microsoft Visual Basic 2010

Visual Basic merupakan salah satu bahasa pemrograman yang andal dan banyak digunakan oleh pengembang untuk membangun berbagai macam aplikasi Windows. Visual basic merupakan aplikasi pemrograman yang menggunakan teknologi .NET Framework. Teknologi .NET Framework merupakan komponen windows yang terintegrasi serta mendukung pembuatan , penggunaan aplikasi, dan halaman WEB. Teknologi .NET Framework mempunyai 2 utama, yaitu CLR (*Common Language Runtime*) dan *Class Library* adalah kelas perintah yang digunakan untuk membangun aplikasi. (Woro Widya ; 2010 : 2)

II.5. SQL Server

SQL Server 2008 adalah teknologi yang mendukung development dan administrasi dari Business Intelligence (BI) Application. SQL Server Reporting and Integration servis adalah element dari BI, tapi inti dari BI tersebut adalah SQL

Server 2008 Analysis Servis (SSAS). Analysis services adalah technology untuk OLAP (Online Analytical Processing) dan data mining. Proses OLAP administration dilakukan di SQL Server Management Studio berupa viewing data, membuat multidimensional Expression (MDX), Data Mining Extensin (DMX) dan XML for analysis (XML/A) dan mendefenisikan role untuk akses security OLAP data. (Kasim Wirama ; 2014 : 42)

II.6. Kamus Data

Kamus data adalah katalog fakta tentang data dan kebutuhan – kebutuhan informasi dari suatu sistem informasi. Kamus data selain digunakan untuk dokumentasi dan mengurangi redudansi, juga dapat digunakan untuk :

1. Memvalidasi diagram aliran data dalam hal kelengkapan dan keakuratan.
2. Menyediakan suatu titik awal untuk mengembangkan layar dan laporan – laporan.
3. Menentukan muatan data yang disimpan dalam *file – file*.
4. Mengembangkan logika untuk proses – proses diagram aliran data.

Kamus data dibuat pada tahap analisis sistem dan digunakan baik pada tahap analisis maupun pada tahap perancangan sistem. Pada tahap analisis sistem, kamus data dapat digunakan sebagai alat komunikasi antara analisis sistem dengan pemakai sistem tentang data yang mengalir di sistem, yaitu tentang data yang masuk ke sistem dan tentang informasi yang dibutuhkan oleh pihak pemakai

sistem. Pada tahap perancangan sistem, kamus data digunakan untuk merancang input, merancang laporan – laporan dan database.

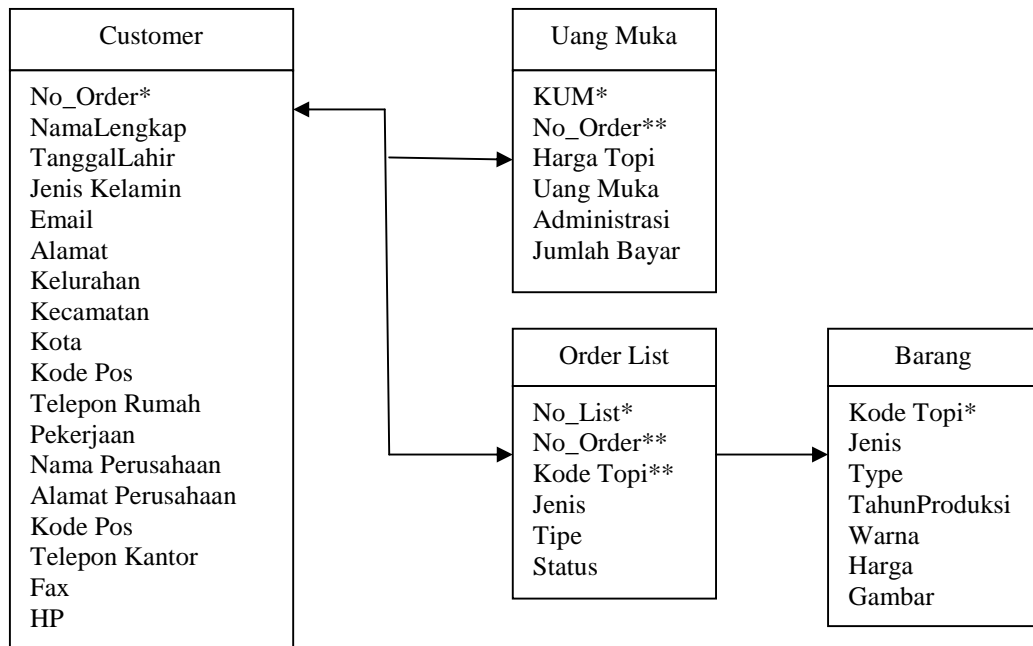
Kamus data mendefinisikan elemen data dengan fungsi sebagai berikut :

1. Menjelaskan arti aliran data dan penyimpanan data dalam DFD.
2. Mendeskripsikan komposisi paket data yang bergerak melalui aliran (misalnya alamat diuraikan menjadi kota, negara dan kode pos)
3. Mendeskripsikan komposisi penyimpanan data.
4. Menspesifikasikan nilai dan satuan yang relevan bagi penyimpanan dan aliran.
5. Mendeskripsikan hubungan detail antar penyimpanan (yang akan menjadi titik perhatian dalam *entity – relationship diagram*). (Jurnal Muhammad Arhami : 2010 ; 7)

II.7. Normalisasi

Normalisasi adalah suatu teknik untuk mengorganisasi data kedalam tabel – tabel untuk memenuhi kebutuhan pemakai didalam suatu organisasi. Normalisasi dilakukan agar basis data yang akan diterapkan dapat digunakan dan dioperasikan dengan efisien, mudah dan tidak mengalami anomali atau keanehan.

Normalisasi dimulai dengan menganalisa tabel dalam bentuk tidak normal, dan berikut contoh normalisasi.



Gambar II.1. Contoh Normalisasi

(Sumber : Iyan Gustiana ; 2011 : 7)

II.8. Entity Relationship Diagram (ERD)

Pada dasarnya ERD (Entity Relationship Diagram) adalah sebuah diagram yang secara konseptual memetakan hubungan antar penyimpanan pada diagram DFD di atas. ERD ini digunakan untuk melakukan permodelan terhadap struktur data dan hubungannya. Penggunaan ERD ini dilakukan untuk mengurangi tingkat kerumitan penyusunan sebuah database yang baik.

Entity dapat berarti sebuah obyek yang dapat dibedakan dengan obyek lainnya. Obyek tersebut dapat memiliki komponen – komponen data (*atribut* atau *field*) yang membuatnya dapat dibedakan dari obyek yang lain. Dalam dunia database entity memiliki attribut yang menjelaskan karakteristik dari entity

tersebut. Ada dua macam atribut yang berperan sebagai kunci primer dan atribut deskriptif. Hal ini berarti setiap entity memiliki himpunan yang diperlukan sebuah primary key untuk membedakan anggota – anggota dalam himpunan tersebut.

Atribut dapat memiliki sifat – sifat sebagai berikut :

1. *Atomic*, atomik adalah sifat dari atribut yang menggambarkan bahwa atribut tersebut berisi nilai yang spesifikasi dan tidak dapat dipecah lagi. Contoh dari sifat atomik adalah field status dari tabel karyawan yang hanya berisi menikah atau singel.
2. *Multivalued*, sifat ini menandakan atribut ini bisa memiliki lebih dari satu nilai untuk tiap entity tertentu. Misalnya adalah field hobi, hobi dari tiap karyawan mungkin dan hampir pasti lebih dari satu. Misalnya karyawan A memiliki hobi membaca, nonton TV dan bersepeda.
3. *Composite*, atribut yang bersifat komposit adalah atribut yang nilainya adalah gabungan dari beberapa atribut yang bersifat atomik. Contohnya adalah atribut alamat yang dapat dipecah menjadi atribut atomik berupa alamat, kode pos, no telepon, dan kota. (Wahana Komputer : 2010 ; 30)

II.9. Unified Modelling Language (UML)

Pengembangan sistem adalah aktivitas manusia. Tanpa adanya kemudahan untuk memahami sistem notasi, proses pengembangan kemungkinan besar akan mengalami kesalahan. UML adalah sistem notasi yang sudah dibakukan di dunia pengembangan sistem, hasil kerjasama dari Grady Booch, James Rumbaugh dan Ivar Jacobson. UML yang terdiri dari serangkaian diagram memungkinkan bagi

sistem analis untuk membuat cetak biru sistem yang komperhesif kepada klien, programmer dan tiap orang yang terlibat dalam proses pengembangan tersebut. Dengan UML akan dapat menceritakan apa yang seharusnya dilakukan oleh sistem bukan bagaimana yang seharusnya dilakukan oleh sebuah sistem.

Menurut Prabowo Pudjo Widodo dan Herlawati (2011 : 6-7) UML diaplikasikan untuk maksud tertentu, biasanya antara lain untuk :

1. Merancang perangkat lunak.
2. Sarana komunikasi antara perangkat lunak dengan proses bisnis.
3. Menjabarkan sistem secara rinci untuk analisa dan mencari apa yang diperlukan sistem.
4. Mendokumentasikan sistem yang ada, proses-proses dan organisasinya.

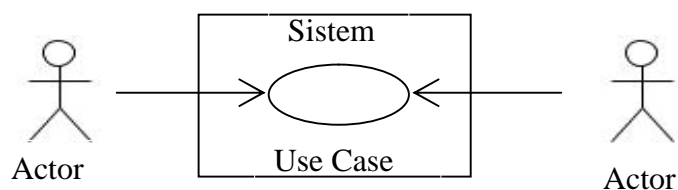
UML adalah bahasa untuk menspesifikasi, memvisualisasi, membangun dan mendokumentasikan bagian dari informasi yang digunakan atau dihasilkan oleh proses pembuatan perangkat lunak, seperti pada pemodelan bisnis dan sistem non perangkat lunak lainnya. UML tidak hanya merupakan sebuah bahasa pemograman visual saja, namun juga dapat secara langsung duhubungkan keberbagai bahasa pemograman seperti JAVA, C++, Visual Basic atau bahkan dihubungkan secara langsung kedalam sebuah *object oriented database*. Begitu juga mengenai pendokumentasian dapat dilakukan seperti *requitments*, arsitektur, *design*, *source*, *project plan*, *tests* dan *prototypes*. UML memiliki 8 tipe diagram, namun pada penulisan skripsi ini penulis akan menggunakan 4 tipe diagram UML yaitu *use case diagram*, *sequence diagram*, *activity diagram* dan *class diagram*.

II.9.1. Tipe Diagram UML

Adapun 8 tipe diagram UML adalah :

1. *Use Case Diagram*

Diagram use case menunjukkan 3 aspek dari sistem, yaitu actor, use case, dan sistem/ sub sistem boundary. Actor mewakili peran orang, sistem yang lain atau alat ketika berkomunikasi dengan use case. Gambar II.1 mengilustrasikan actor, use case dan boundary.



Gambar II.2 Use Case Model

Sumber : Prabowo Pudjo Widodo dan Herlawati (2011)

2. *Class Diagram*

Class diagram menggambarkan struktur dan deskripsi class, package dan objek beserta hubungan satu sama lain seperti containment, pewarisan, asosiasi dan lain-lain.

3. *Statechart Diagram*

Diagram Statechart menggambarkan semua state yang dimiliki oleh suatu objek dari suatu class dan keadaan yang menyebabkan state berubah.



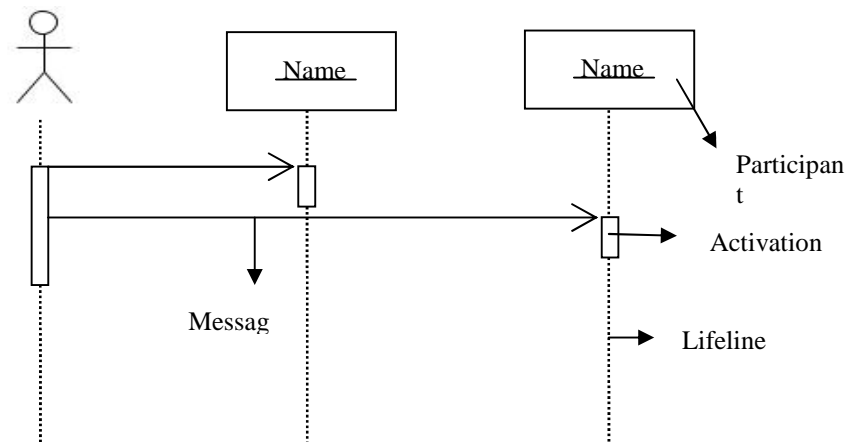
Gambar II.3. Simbol Diagram *Statechart*

Sumber : Prabowo Pudjo Widodo dan Herlawati (2011)

Pada gambar II.2. bentuk lingkaran solid memaparkan bagaimana objek dibentuk atau diawali, tanda panah (*transition*) menjelaskan bagaimana sebuah kejadian yang memicu sebuah state objek dengan cara memperbaharui satu atau lebih nilai atributnya, dan *double round* memaparkan bagaimana sebuah objek dibentuk dan dihancurkan.

4. *Sequence Diagram*

Sequence diagram digunakan untuk menggambarkan perilaku pada sebuah skenario. Diagram ini menunjukkan sejumlah contoh obyek dan pesan yang diletakkan diantara obyek-obyek dalam *use case*.



Gambar II.4. Simbol-Simbol *Sequence Diagram*

Sumber : Prabowo Pudjo Widodo dan Herlawati (2011)

5. *Collaboration Diagram*




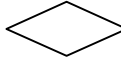


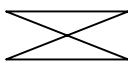

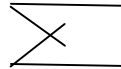

Menggambarkan kolaborasi dinamis seperti *sequence diagram*.

Dalam menunjukkan pertukaran pesan, *collaboration diagram* dengan menggunakan objek dan hubungannya.

6. *Activity Diagram*

Menggambarkan rangkaian aliran dari aktivitas, digunakan untuk mendeskripsikan aktivitas yang dibentuk dalam suatu operasi sehingga dapat juga digunakan untuk aktivitas lainya seperti *Use Case* atau interaksi.

Tabel II.7. Simbol Activity Diagram

Gambar	Nama
	Titik awal
	Titik akhir
	<i>Activity</i>
	Pilihan untuk pengambilan keputusan
	<i>Fork</i> , digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau untuk menggabungkan dua kegiatan parallel menjadi satu
	<i>Rake</i> , menunjukkan adanya dekomposisi
	Tanda waktu
	Tanda pengiriman
	Tanda penerimaan
	<i>Flow final</i>

Sumber : Prabowo Pudjo Widodo dan Herlawati (2011)

7. *Component Diagram*

Component diagram menggambarkan struktur dan hubungan antara komponen piranti lunak, termasuk ketergantungan. Komponen dapat juga berupa interface, yaitu kumpulan layanan yang disediakan sebuah komponen untuk komponen lainnya.

8. *Deployment Diagram*

Deployment diagram menggambarkan arsitektur fisik dari perangkat keras dan perangkat lunak system, menunjukkan hubungan computer dengan perangkat satu sama lain dan jenis hubungannya.