

## BAB II

### LANDASAN TEORI

#### 2.1. SISTEM

Pengertian dari sistem sangat luas dan beranekaragam, sehingga timbul berbagai definisi dan istilah tentang sistem :

Menurut *Robert G. Murdick, et al* :

*Sistem adalah seperangkat elemen yang membentuk suatu kegiatan atau suatu prosedur / bagan pengolahan yang mencari suatu tujuan dengan mengoperasikan data atau barang pada waktu tujuan tertentu untuk menghasilkan informasi (Kadir, 2013).*

Sedangkan menurut *Davis et, al* :

*Sistem merupakan kumpulan dari elemen-elemen (sub sistem) yang secara bersama-sama membentuk satu keastuan dan saling berinteraksi dalam menacapai tujuan (Kadir, 2013)*

Terdapat dua kelompok pendekatan didalam mendefinisikan sistem yaitu :

1. Menekankan pengertian sistem pada prosedur dan mendefinisikan sistem sebagai suatu jaringan kerja dari prosedur-prosedur yang saling berhubunga, berkumpul bersama-sama untuk melakukan suatu kegiatan dan menyelesaikan masalah.
2. Sistem sebagai kumpulan elemen-elemen yang berinteraksi, berhubungan untuk mencapai tujuan tertentu.

Pendekatan sistem merupakan jaringan kerja dari prosedur yang lebih menekankan pada urutan operasi sistem. Sedangkan pendekatan sistem yang menekankan pada komponen elemen akan lebih mudah dalam mempelajari suatu sistem untuk tujuan analisis dan perancangan suatu sistem.

Suatu sistem mempunyai karakteristik atau sifat-sifat tertentu yang antara lain adalah sebagai berikut :

1. **Komponen Sistem**

Salah satu karakteristik sistem yang berupa sub sistem atau bagian-bagian sistem. Setiap sub sistem memiliki sifat-sifat dari sub sistem untuk menjalankan suatu fungsi tertentu dan mempengaruhi proses secara keseluruhan.

2. **Batas Sistem**

Merupakan daerah yang membatasi antara sistem dengan sistem lainnya atau lingkungan luarnya. Batas sistem juga mungkin suatu sistem dipandang sebagai satu kesatuan dan menunjukkan ruang lingkup dari sistem tersebut.

3. **Lingkungan Luar Sistem**

Segala sesuatu yang berada diluar batas dari sistem yang mempengaruhi operasi baik itu menguntungkan operasi sistem yang merupakan energi dari sistem yang harus dikendalikan dan ditahan sehingga tidak mengganggu atau merusak sistem.

#### 4. Penghubung Sistem

Merupakan media penghubung anantara satu sub sistem dengan sub sistem lainnya atau bisa dilakukan untuk mengintegrasikan antara sub sistem yang satu dengan sub sistem lainnya membentuk satu kesatuan.

#### 5. Masukan Sistem

Masukan dari sistem adalah data yang dimasukkan kedalam sistem yang diproses untuk mendapat satu keastuan atau informasi.

#### 6. Keluaran Sistem

Keluaran sistem adalah data yang diolah serta di klasifikasikan menjadi keluaran yang berguna atau menjadi informasi yang dibutuhkan.

#### 7. Pengolah Sistem

Suatu sistem dapat mempunyai suatu bagian pengolahan yang akan merubah suatu masukan menjadi keluaran yang dibentuk.

#### 8. Sasaran Sistem

Suatu sistem mempunyai maksud tertentu yaitu tujuan atas sasaran, dimana yang menentukan sekali masukan serta keluaran sistem yang mengenai pada sasaran atau tujuan yang dimaksud.

## 2.2. SISTEM PENDUKUNG KEPUTUSAN (*Decision Support System*)

### 2.2.1. Definisi Sistem Pendukung Keputusan

Sistem pendukung keputusan menurut Alters Keen merupakan sistem informasi interaktif yang menyediakan informasi, pemodelan, dan pemanipulasian data. Sistem itu digunakan untuk membantu pengambilan keputusan dalam situasi semistruktur dan situasi yang tidak terstruktur, dimana tak seorang pun tahu secara pasti bagaimana keputusan seharusnya dibuat (Kusrini, 2010).

Definisikan oleh Michael S.Scott Morton sebagai sistem berbasis interaktif, yang membantu para pengambil keputusan untuk menggunakan data dan berbagai model untuk memecahkan masalah-masalah tidak terstruktur (Tuban, 2010)

Menurut Sprague konsep sistem pendukung keputusan pertama kali diperkenalkan pada awal tahun 1970-an oleh Michael S.Scott Morton dengan istilah *Management Decision System*.

Pada dasarnya Sistem Pendukung Keputusan (SPK) dirancang untuk mendukung seluruh tahap pengambilan keputusan mulai dari mengidentifikasi masalah, memilih data yang relevan, menentukan pendekatan yang digunakan dalam proses pengambilan keputusan, sampai mengevaluasi pemilihan alternatif.

### 2.2.2. Karakteristik Sistem Pendukung Keputusan

Menurut Suryadi dan Ramdhani (2010) peranan Sistem pendukung Keputusan (SPK), dalam konteks keseluruhan sistem informasi ditujukan untuk memperbaiki kinerja melalui aplikasi teknologi informasi. Terdapat beberapa karakteristik dasar Sistem Pendukung Keputusan (SPK) yang efektif, yaitu sebagai berikut :

1. Mendukung proses pengambilan keputusan, menitik beratkan pada *management of perception*.
2. Adanya interface manusia-msin dimana manusia (user) tetap mengontrol proses pengambilan keputusan.
3. Mendukung pengambilan keputusan untuk membahas masalah-masalah terstruktur, semi terstruktur, dan tidak terstruktur.
4. Output ditujukan untuk personil organisasi dalam semua tingkatan.
5. Memiliki subsistem - subsistem yang terintegrasi sedemikian rupa sehingga dapat berfungsi sebagai kesatuan sistem.
6. Membutuhkan struktur data komprehensif yang dapat melayani kebutuhan informasi seluruh tingkatan manajemen.
7. Pendekatan *easy to use*. Ciri suatu SPK yang efektif adalah kemudahannya untuk digunakan, dan memungkinkan keleluasaan pemakai untuk memilih atau mengembangkan pendekatan-pendekatan baru dalam membahas sistem yang dihadapi.

8. Kemampuan sistem beradaptasi secara cepat, dimana pengambil keputusan dapat menghadapi masalah-masalah baru, dan pada saat yang sama dapat menanganinya dengan cara mengadaptasikan sistem terhadap kondisi-kondisi perubahan yang terjadi.

### **2.3. TOPSIS (*Technique For Order Preference by Similiarity to Ideal Solution*)**

TOPSIS adalah salah satu metode pengambilan keputusan yang multikriteria yang pertama kali diperkenalkan oleh *Yoon* dan *Hwang* (1981). TOPSIS menggunakan prinsip bahwa alternatif yang terpilih harus mempunyai jarak terdekat dari solusi ideal positif dan terjauh dari solusi ideal negatif dari sudut pandang geometris dengan menggunakan jarak *Euclidean* untuk menentukan kedekatan relatif dari suatu alternatif dengan solusi optimal. Solusi ideal positif didefinisikan sebagai jumlah dari seluruh nilai terbaik yang dapat dicapai untuk setiap atribut, sedangkan solusi ideal negatif terdiri dari seluruh nilai terburuk yang dicapai untuk setiap atribut.

TOPSIS mempertimbangkan keduanya, jarak terhadap solusi ideal positif dan jarak terhadap solusi ideal negatif dengan mengambil kedekatan relatif terhadap solusi ideal positif. Berdasarkan perbandingan terhadap jarak relatifnya, susunan prioritas alternatif bisa dicapai. Metode ini banyak digunakan untuk menyelesaikan pengambilan keputusan. Hal ini disebabkan

konsepnya sederhana, mudah dipahami, komputasinya efisien, dan memiliki kemampuan mengukur kinerja relatif dari alternatif-alternatif keputusan.

Secara umum, prosedur TOPSIS mengikuti langkah-langkah sebagai berikut :

- 1) Membuat matriks keputusan yang ternormalisasi.
- 2) Membuat matriks keputusan yang ternormalisasi terbobot.
- 3) Menentukan matriks solusi ideal positif dan matriks solusi ideal negatif.
- 4) Menentukan jarak antara nilai setiap alternatif dengan matriks solusi ideal positif dan matriks solusi ideal negatif.
- 5) Menentukan nilai preferensi untuk setiap alternatif

TOPSIS membutuhkan rating kinerja setiap alternatif  $A_i$  pada setiap kriteria  $C_j$  yang ternormalisasi, yaitu :

$$R_{ij} = \frac{X_{ij}}{\sqrt{\sum_{i=1}^m X_{ij}^2}} ; \text{ dengan } i=1,2,\dots,m; \text{ dan } j=1,2,\dots,n \quad (1)$$

Solusi ideal positif  $A^+$  dan solusi ideal negatif  $A^-$  dapat ditentukan berdasarkan rating bobot ternormalisasi ( $y_{ij}$ ) sebagai :

$$y_{ij} = w_j r_{ij} ; \text{ dengan } i=1,2,\dots,m; \text{ dan } j=1,2,\dots,n. \quad (2)$$

$$A^+ = (y_1^+, y_2^+, \dots, y_n^+) \quad (3)$$

$$A^- = (y_1^-, y_2^-, \dots, y_n^-) \quad (4)$$

dengan

$$y_j^+ = \begin{cases} \max_i y_{ij} & ; \text{jika } j \text{ adalah atribut keuntungan} \\ \min_i y_{ij} & ; \text{jika } j \text{ adalah biaya} \end{cases} \quad (5)$$

$$y_j^- = \begin{cases} \min_i y_{ij} & ; \text{jika } j \text{ adalah atribut keuntungan} \\ \max_i y_{ij} & ; \text{jika } j \text{ adalah biaya} \end{cases} \quad (6)$$

$j=1,2,\dots,n$ .

Jarak antara alternatif  $A_i$  dengan solusi ideal positif dirumuskan sebagai :

$$D_i^+ = \sqrt{\sum_{j=1}^n (y_i^+ - y_{ij})^2} ; i=1,2,\dots,m. \quad (7)$$

Jarak antara alternatif  $A_i$  dengan solusi ideal negatif dirumuskan sebagai :

$$D_i^- = \sqrt{\sum_{j=1}^n (y_{ij} - y_i^-)^2} ; i=1,2,\dots,m. \quad (8)$$

Nilai preferensi untuk setiap alternatif ( $V_i$ ) diberikan sebagai :

$$V_i = \frac{D_i^-}{D_i^+ + D_i^-} ; i=1,2,\dots,m. \quad (9)$$

Nilai  $V_i$  yang lebih besar menunjukkan bahwa alternatif  $A_i$  lebih dipilih.

## 2.4. APLIKASI

Aplikasi adalah program yang digunakan untuk melakukan pekerjaan tertentu, misalkan aplikasi untuk menghitung gaji suatu perusahaan dan aplikasi untuk kasir dalam swalayan. Biasanya aplikasi dibuat oleh perusahaan atas permintaan seseorang/lembaga perusahaan-perusahaan, baik besar maupun kecil yang melayani pembuatan aplikasi



untuk keperluan perusahaan, lembaga ataupun pereorangan (Maryono, 2011).

## **2.5. BANTUAN SISWA MISKIN (BSM)**

Salah satu upaya pemerintah dalam meningkatkan akses pendidikan kepada masyarakat khususnya pada siswa yang berasal dari keluarga kurang mampu adalah digulirkannya program Bantuan Siswa Miskin (BSM). Program Bantuan Siswa Miskin (BSM) adalah bantuan dari Pemerintah berupa sejumlah uang tunai yang diberikan secara langsung kepada siswa dari semua Jenjang Pendidikan (SD/MI, SMP/MTs, SMA/SMK/MA/MAK) yang berasal dari keluarga miskin dan rentan miskin sesuai dengan kriteria yang telah ditetapkan.

Penetapan sasaran Program BSM, dari yang semula melalui sekolah, telah diubah menjadi Penetapan Sasaran Berbasis Rumah Tangga melalui pemberian Kartu Perlindungan Sosial (KPS) kepada rumah tangga miskin dan rentan kemiskinan. Selanjutnya rumah tangga yang memiliki anak-anak berusia sekolah dapat membawa KPS tersebut ke sekolah untuk dicalonkan sebagai Penerima Manfaat Program BSM. Kartu ini diberikan dengan tujuan untuk memperbaiki ketepatan sasaran penerima Program BSM agar menjangkau anak-anak sekolah yang berasal dari rumah tangga miskin dan rentan sesuai kuota dan pagu anggaran yang tersedia.

Tujuan pemberian Bantuan Siswa Miskin (BSM) adalah mengamankan program pemerintah dalam penuntasan wajib belajar dua

belas tahun (Pendidikan Menengah Universal). Secara khusus program BSM ini bertujuan :

1. Menghilangkan halangan siswa miskin berpartisipasi untuk bersekolah dengan membantu siswa miskin untuk memperoleh akses pelayanan pendidikan yang layak.
2. Mencegah angka putus sekolah dan menarik siswa miskin untuk bersekolah.
3. Membantu siswa miskin memenuhi kebutuhan dalam kegiatan pembelajaran.
4. Mendukung penuntasan wajib belajar pendidikan dasar sembilan tahun bahkan hingga tingkat menengah atas.

Kriteria umum siswa penerima Bantuan Siswa Miskin (BSM) adalah :

1. Siswa anggota Rumah Tangga penerima Kartu Perlindungan Sosial (KPS) yang telah terdaftar sebagai penerima BSM tahun 2013.
2. Siswa anggota Rumah Tangga penerima Kartu Perlindungan Sosial (KPS) yang belum terdaftar dan belum menerima BSM tahun 2013.

Selain kriteria diatas dan apabila kuota masih tersedia, Kepala Madrasah bersama dengan Komite Madrasah dapat mengusulkan nama siswa lain yang dianggap pantas dan berhak mendapatkan BSM tetapi tidak mendapatkan kartu dengan kriteria sebagai berikut :

1. Orang tua siswa terdaftar sebagai Peserta PKH (Program Keluarga Harapan).

2. Siswa yang berasal dari panti sosial/panti asuhan yang dikelola oleh Kementerian Sosial.
3. Siswa korban musibah bencana alam.
4. Rumah tangga pemegang Surat Keterangan Tidak Mampu (SKTM) dari Kelurahan/Desa.
5. Siswa terancam putus sekolah karena kesulitan biaya.
6. Yatim dan atau Piatu.
7. Nilai Rapor, Penghasilan, Tanggungan, Jenis Dinding, Jenis Lantai, dll
8. Pertimbangan lain (misalkan kelainan fisik, korban musibah berkepanjangan dan siswa berasal dari rumah tangga miskin dan memiliki lebih dari 3 (tiga) orang bersaudara yang berusia dibawah 18 tahun).

## **2.6. KOMPUTER**

Komputer adalah alat elektronik yang mampu melaksanakan tugas antara lain menerima input, memproses input sesuai dengan programnya, menyimpan perintah-perintah dan hasil pengolahan, menyediakan output dalam bentuk informasi (Blissmer, 2011).

Komputer adalah sistem elektronik untuk memanipulasi data yang cepat dan tepat serta dirancang dan diorganisasikan agar secara otomatis menerima dan menyimpan data input, memprosesnya dan menghasilkan output berdasarkan instruksi-instruksi yang telah tersimpan dalam memori. (Sanders, 2011).

Komputer merupakan suatu alat elektronik dengan kecepatan tinggi yang mampu melaksanakan perhitungan dan operasi yang logis serta menyimpan dan melaksanakan serangkaian operasi tanpa campur tangan manusia.

Komputer adalah serangkaian mesin elektronik yang terdiri dari ribuan bahkan jutaan komputer yang dapat saling bekerja sama, serta membentuk sebuah sistem kerja yang rapi dan teliti. (Tutang, 2011).

Adapun perangkat-perangkat komputer adalah sebagai berikut :

#### **1. Perangkat Keras (*Hardware*)**

Perangkat keras digunakan peralatan pada sistem komputer yang secara fisik dapat dilihat dan dipegang. Bagian-bagian pokok perangkat keras yaitu :

- a. Unit Masukan (*Input Device*) yaitu alat yang digunakan untuk menerima masukan berupa data atau program.
- b. Unit Pemroses (*Central Processing Unit*) yaitu alat dimana instruksi-instruksi program diproses untuk mengolah data.
- c. Unit Penyimpanan (*Secondary Storage*).
- d. Berbeda dengan memori, secondary storage bersifat lebih tetap.
- e. Unit Keluaran (*Output Device*) yaitu alat untuk mengeluarkan hasil proses komputer.

## 2. Perangkat Lunak (*Software*)

Suatu program yang dibuat oleh pembuat program untuk menjalankan perangkat keras komputer ada tiga bagian perangkat software ini yaitu :

### a. Sistem Operasi (*Operating System*)

Sistem Operasi (*Operating System*) adalah program yang ditulis untuk mengendalikan dan mengorganisasikan kegiatan dari seluruh sistem.

### b. Perangkat Lunak Bahasa

Perangkat lunak bahasa yaitu program-program yang digunakan untuk menterjemahkan instruksi-instruksi yang ditulis dalam bahasa pemrograman, kedalam bahasa mesin agar dapat dimengerti.

### c. Perangkat Lunak Aplikasi (*Aplication Software*)

Merupakan program yang ditulis dan diterjemahkan oleh perangkat bahasa yaitu program untuk menyelesaikan suatu permasalahan tertentu.

## 3. Pengguna (*Brainware*)

Brainware adalah pengguna komputer atau orang-orang yang berhubungan dengan komputer, brainware dibedakan menjadi empat yaitu :

a. Analisis Komputer

Seorang analisi adalah orang yang bertanggung jawab pada pembuatan, perencanaan suatu aplikasi tertentu secara keseluruhan.

b. Programmer

Merupakan orang yang bekerja membuat aplikasi komputer, menyusun instruksi-instruksi untuk komputer, menguji program-program serta menyiapkan dokumentasi.

c. Operator

Merupakan orang yang bertugas mengoperasikan program aplikasi yang disusun oleh seseorang programmer, dengan mengikuti instruksi yang sebelumnya telah dituangkan ke dalam pedoman menjalankan program.

d. Librarian

Petugas yang berwenang pada pemeliharaan dan penyimpanan program-program, file instruksi atas catatan komputer lainnya.

## 2.7. PHP

Pada awalnya PHP merupakan singkatan dari *Personal Home Page tools*, yang gunanya untuk memonitor pengunjung web. PHP mula-mula dikembangkan oleh Rasmus Lerdorf. Kemudian istilah PHP mengacu pada *Jypertext Perprocessor*. PHP merupakan bahasa berbentuk skrip yang

ditempatkan dalam server dan di proses di server. Hasilnya akan dikirimkan ke client, tempat pemakai menggunakan browser. PHP dikenal sebagai sebuah bahasa scripting, yang menyatu dengan tag-tag HTML, dieksekusi di server, dan digunakan untuk membuat halaman web yang dinamis seperti halnya *Active Server Pages (ASP)* atau *Java Server Pages (JSP)*. PHP merupakan sebuah software open source.

Kelebihan PHP dari bahasa pemrograman lain yaitu :

1. Bahasa pemrograman PHP adalah sebuah bahasa script yang tidak melakukan sebuah kompilasi dalam penggunaannya.
2. Web server yang mendukung PHP dapat ditemukan dimana-mana dari mulai *apache, IIS, Lighted* , hingga *Xitami* dengan konfigurasi yang relatif mudah.
3. Dalam sisi pengembangan lebih mudah, karena banyaknya milis-milis dan developer yang siap membantu dalam pengembangan.
4. Dalam sisi pemahaman, PHP adalah bahasa scripting yang paling mudah karena memiliki referensi yang banyak.

## **2.8. BASIS DATA**

Basis Data adalah kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan yang tidak perlu. (Fathansyah, 2010).

Basis data mempunyai berbagai sumber data dalam pengumpulan data, bervariasi derajat interaksi kejadian dari dunia nyata, dirancang dan

dibangun agar dapat digunakan oleh beberapa user untuk berbagai kepentingan. (Waliyanto, 2010).

## 2.9. MYSQL

Merupakan sebuah perangkat lunak sistem manajemen basis data SQL/ DBMS (*Database Management System*) yang *multithread*, *multi-user* dan sekitar 6 juta instalasi diseluruh Indonesia. Didistribusikan secara gratis dibawah lisensi GPL (*General Public License*). Dimana setiap orang bebas untuk menggunakan MySQL, namun tidak boleh dijadikan produk turunan yang bersifat komersial.

Keistimewaan MySQL yakni :

1. Portabilitas MySQL dapat berjalan stabil pada berbagai sistem operasi seperti Windows, Linux, FreeBSD, Mac Os X Server, Solaris, Amiga, dan masih banyak lagi.
2. Open Source MySQL didistribusikan secara open source, dibawah lisensi GPL sehingga dapat digunakan secara cuma -cuma.
3. Multiuser MySQL dapat digunakan oleh beberapa user dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.
4. Performance tuning, MySQL memiliki kecepatan yang menakjubkan dalam menangani query sederhana, dengan kata lain dapat memproses lebih banyak SQL per satuan waktu.



5. Jenis kolom MySQL memiliki tipe kolom yang sangat kompleks, seperti signed / unsigned integer, float, double, char, text, date, timestamp, dan lain-lain.
6. Perintah dan fungsi MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah Select dan Where dalam perintah (query).
7. Keamanan MySQL memiliki beberapa lapisan keamanan seperti level subnetmask, nama host, dan izin akses user dengan sistem perizinan yang mendetail serta sandi terenkripsi.

## **2.10. DESAIN SISTEM**

### **2.10.1. Data Flow Diagram (DFD)**

Diagram yang menggunakan notasi simbol untuk menggambarkan arus data sistem. (Jogiyanto Hartono, 2005, 701).

Menurut Jogiyanto Hartono, tahun 2005 dalam bukunya Basis Data ada beberapa simbol digunakan pada DFD untuk mewakili :

#### **1. Kesatuan Luar (External Entity)**

Kesatuan luar (external entity) merupakan kesatuan (entity) di lingkungan luar sistem yang dapat berupa orang, organisasi, atau sistem lain yang berada pada lingkungan luarnya yang memberikan input atau menerima output dari sistem.

## 2. Arus Data (Data Flow)

Arus Data (Data Flow) di DFD diberi simbol suatu panah. Arus data ini mengalir di antara proses, simpan data dan kesatuan luar. Arus data ini menunjukkan arus dari data yang dapat berupa masukan untuk sistem atau hasil dari proses sistem.

## 3. Proses (Process)

Proses (Process) menunjukkan pada bagian yang mengubah input menjadi output, yaitu menunjukkan bagaimana satu atau lebih input diubah menjadi beberapa output. Setiap proses mempunyai nama, nama dari proses ini menunjukkan apa yang dikerjakan proses.

## 4. Data Store

Merupakan simpanan dari data yang dapat berupa suatu file atau database pada sistem komputer.

### **2.10.2. Entitas Relationship Diagram**

Entitas Relationship Diagram atau ERD, adalah mendokumentasikan data perusahaan dengan mengidentifikasi jenis dan hubungannya. Komponen-komponen ERD yaitu :

#### 1. Entitas

Jenis entitas (Entity Type) dapat berupa suatu elemen lingkungan, sumber daya, atau transaksi yang begitu pentingnya bagi perusahaan sehingga di dokumentasikan

dengan data jenis entitas di dokumentasikan dengan symbol persegi panjang.

## 2. Hubungan

Hubungan adalah suatu asosiasi yang ada antara dua jenis entitas. Hubungan digambarkan dengan bentuk belah ketupat. Tiap belah ketupat diberi label kata kerja.

## 3. Atribut

Atribut adalah karakteristik dari suatu entitas. Atribut-atribut tersebut sebenarnya adalah elemen-elemen data dan masing-masing diberikan satu nilai tunggal, yang disebut nilai atribut digambarkan dalam bentuk elips.

### **2.11. UML (*Unified Modeling Language*)**

*Unified Modeling Language* (UML) merupakan alat merancang perangkat lunak, sarana komunikasi antara perangkat lunak dengan, menjabarkan sistem secara rinci untuk analisa dan mencari apa yang diperlukan sistem, mendokumentasikan sistem yang ada, proses-proses dan organisasinya (Herlawati, 2011). *Unified Modeling Language* (UML) adalah salah satu bentuk language atau bahasa, menurut pencetusnya UML di definisikan sebagai bahasa visual untuk menjelaskan, memberikan spesifikasi, merancang, membuat model, dan mendokumentasikan aspek-aspek dari sebuah sistem.

Salah satu cara untuk mengatur diagram UML adalah dengan menggunakan *view*. *View* adalah kumpulan diagram yang menggambarkan aspek yang sama dari proyek. *View* mempunyai 3 pelengkap, yaitu *Static View*, *Dynamic View*, dan *Functional View*.

#### **2.11.1. *Static View***

*Static View* termasuk diagram yang memberikan gambaran dari unsur-unsur dari sistem tetapi tidak memberitahu bagaimana elemen akan berperilaku. Hal ini sangat mirip *Blueprint*. *Blueprint* itu komprehensif, tetapi mereka hanya menunjukkan apa yang tetap diam, maka disebut *Static View*. *Static View* dibentuk oleh dua diagram, yaitu *Class Diagram* dan *Object Diagram*.

#### **2.11.2. *Dynamic View***

Pada *Dynamic View* meliputi diagram yang mengungkapkan bagaimana benda berinteraksi dengan satu sama lain dalam respon terhadap lingkungan. Ini termasuk *Sequence Diagram* dan *Collaboration Diagram*, yang kolektif disebut sebagai diagram interaksi. Mereka secara khusus dirancang untuk menjelaskan bagaimana benda berbicara satu sama lain. Ini juga mencakup *Statechart Diagram*, yang menunjukkan bagaimana dan mengapa perubahan objek dari waktu ke waktu dalam menanggapi lingkungan.





### 2.11.3. Functional View

*Functional View* terbentuk oleh *Use Case Diagram* dan *Activity Diagram*.

#### 2.11.3.1. 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. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Syarat penamaan pada *use case* adalah nama didefinisikan sesimpel mungkin dan dapat dipahami. Ada dua hal utama pada *use case* yaitu pendefinisian apa yang disebut aktor dan *use case*. Simbol-simbol yang ada pada diagram *use case* dan ditunjukkan pada tabel 2.1.

Tabel 2.1 Simbol Use Case Diagram

Simbol	Keterangan
	<i>Actor</i> : Sebuah peran yang dimainkan oleh seseorang, sistem, atau perangkat yang memiliki saham dalam keberhasilan operasi dari sistem.
	<i>Use Case</i> : Untuk mengungkapkan tujuan bahwa sistem harus dicapai.
	<i>Association</i> : Mengidentifikasi interaksi antara aktor dan <i>Use Case</i>
	<i>Dependency</i> : Mengidentifikasi hubungan komunikasi antara dua <i>Use Case</i>










### 2.11.3.2. Activity Diagram

Diagram aktivitas atau *activity diagram* menggambarkan *workflow* atau aktivitas dari sebuah sistem atau proses bisnis. Yang perlu diperhatikan disini bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan sistem. Diagram aktivitas juga banyak digunakan untuk mendefinisikan hal-hal berikut :

1. Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan,
2. Urutan atau pengelompokkan 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

Berikut simbol-simbol yang ada pada diagram aktivitas dan ditunjukkan dengan tabel simbol activity diagram pada tabel 2.2

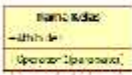


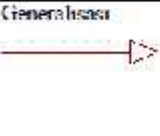
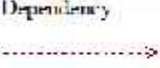

Tabel 2.2 Simbol activity diagram

Simbol	Keterangan
	Titik Awal
	Titik Akhir
	Activity
	Pilihan untuk pengambilan keputusan
	Fork : Untuk menunjukkan kekuatan yang dilakukan secara parallel
	Join : Menunjukkan adanya dekomposisi
	Tanda Waktu
	Tanda Penetrasian
	Akhir Akhir (Flow Final)

### 2.11.3.3. Class Diagram

Kelas Diagram terdiri dari tiga kompartemen (ruang persegi panjang) yang mengandung informasi yang berbeda diperlukan untuk menjelaskan sifat-sifat satu jenis objek. Simbol-simbol yang digunakan dalam class diagram. Berikut simbol-simbol yang digunakan dalam *class diagram* dan ditunjukkan pada tabel 2.3

Tabel 2.3 Simbol Class Diagram







Simbol	Deskripsi
	Kelas pada struktur system
	Paket/package merupakan sebuah bungkus dari satu atau lebih kelas (kumpulan kelas)
	Asosiasi merupakan hubungan antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
	Generalisasi merupakan hubungan generalisasi dan spesialisasi (umum khusus) antara dua kelas dimana fungsi yang satu adalah fungsi yang lebih umum dari fungsi yang lainnya
	Dependency merupakan hubungan antar kelas yang saling bergantung, membutuhkan satu sama lain
	Agregasi merupakan hubungan antar kelas dimana satu kelas merupakan semua bagian dari kelas-kelas yang lain.

### 2.11.3.4. 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 diinstansiasi menjadi objek itu.

Banyaknya diagram sekuen yang harus digambar adalah sebanyak pendefinisian *use case* yang memiliki proses sendiri atau yang penting semua *use case* yang telah didefinisikan interaksi jalannya pesan sudah dicakup pada diagram sekuen sehingga semakin banyak *use case* yang didefinisikan maka diagram sekuen yang harus dibuat juga semakin banyak. Berikut adalah simbol-simbol yang ada pada *Sequence Diagram* yang ditunjukkan pada tabel 2.4.

Tabel 2.4 Simbol Sequence Diagram

Nama	Simbol	Keterangan
Objek dan kelas		Objects, mewakili peserta
<i>Lifeline</i>		<i>Lifeline</i> merupakan siklus hidup sebuah pesan/message berdasarkan waktu
<i>Message</i>		Pesan/message merupakan komunikasi antara objek yang satu dengan objek yang lainnya
<i>Return message</i>		<i>Return message</i> merupakan balasan/hasil yang berisi nilai dari sebuah objek yang meminta (mengirim pesan)
<i>Self message</i>		<i>Self message</i> merupakan pesan dari sebuah objek kepada objek itu sendiri untuk melakukan suatu aksi
<i>Return self message</i>		Balasan/hasil dari <i>self message</i> yang berisi suatu nilai kepada objek itu sendiri