

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Analisis**

Analisis menurut kamus besar bahasa Indonesia adalah penyelidikan terhadap suatu peristiwa meliputi (karangan atau perbuatan, dan sebagainya) untuk mengetahui keadaan sebenarnya (sebab-musabab, duduk perkaranya, dan sebagainya). sehingga dengan melakukan suatu analisis dapat menguraikan pokok masalah dari berbagai kemungkinan yang bisa terjadi.

Menurut Yakub (2012), “Analisa sistem dapat diartikan sebagai suatu proses untuk memahami sistem yang ada, dengan menganalisa jabatan dan uraian tugas (*business users*), proses bisnis (*business proses*), ketentuan atau aturan (*business rule*), masalah dan mencari solusinya (*business problem and business soulution*), dan rencana-rencana perusahaan (*business plan*)”.

Menurut Abdul Kadir (2014), “Analisa Sistem adalah proses untuk menentukan hal-hal detail tentang yang akan dikerjakan oleh sistem yang diusulkan (dan bukan bagaimana caranya). Tahapan analisa sistem dilakukan untuk mengembangkan sistem yang sudah ada atau mengatasi masalah-masalah yang belum tertangani”.

Berdasarkan definisi-definisi tersebut dapat disimpulkan bahwa analisis adalah suatu proses cara kerangka berfikir untuk mengetahui suatu hal atau permasalahan yang akan diselesaikan.

## 2.2 Perancangan

Pengertian perancangan menurut Ladjamudin (2013), tahapan perancangan desain memiliki tujuan untuk mendesain sistem baru yang dapat menyelesaikan masalah-masalah yang dihadapi perusahaan yang diperoleh dari pemilihan alternatif sistem yang terbaik. Kegiatan yang dilakukan dalam tahap perancangan ini meliputi perancangan *output*, *input*, dan *file*.

### 1) Perancangan Keluaran (*Output*)

Perancangan keluaran bertujuan menentukan keluaran-keluaran yang akan digunakan oleh sistem. Keluaran tersebut berupa tampilan-tampilan layar, dan juga format dan frekuensi laporan yang diperlukan.

### 2) Perancangan Masukan (*Input*)

Perancangan masukan bertujuan menentukan data-data masukan, yang akan digunakan untuk mengoperasikan sistem. Data-data masukan tersebut dapat berupa formulir-formulir, faktur, dan lain-lain yang berfungsi memberikan data masukan bagi pemrosesan sistem. Pada tahap ini perlu juga ditentukan format data masukan agar sesuai dengan kebutuhan sistem.

## 2.3 Sistem

Pengertian sistem Menurut Tata Sutabri (2012), “Suatu sistem pada dasarnya adalah sekelompok unsur yang erat hubungannya satu dengan yang lain, yang berfungsi bersama-sama untuk mencapai tujuan tertentu”.

Menurut Azhar Susanto (2013), “Sistem adalah kumpulan/group dari sub sistem/bagian/komponen apapun baik fisik ataupun non fisik yang saling

berhubungan satu sama lain dan bekerja sama secara harmonis untuk mencapai satu tujuan tertentu”.

Berdasarkan dari definisi-definisi diatas dapat disimpulkan bahwa sistem adalah sekelompok unsur atau komponen-komponen atau sub bagian sistem yang saling bekerjasama untuk mencapai tujuan tertentu.

### **2.3.1 Klasifikasi Sistem**

#### **a. Konsep Dasar Sistem**

Menurut Hutahaean (2015), ”Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan,berkumpul bersama-sama untuk melakukan kegiatan atau untuk melakukan sasaran yang tertentu”.

#### **b. Karakteristik Sistem**

Menurut Hutahaean (2015), ”suatu sistem dapat dikatakan sebagai sistem yang baik apabila memiliki karakteristik-karakteristik tertentu”. Karakteristik sistem (yang dimaksud yaitu:

##### **1) Komponen Sistem (*Components*)**

Komponen-komponen sistem tersebut dapat berupa suatu bentuk subsistem. Setiap subsistem memiliki sifat dari sistem yang menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan.

##### **2) Batasan Sistem (*Boundary*)**

Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem yang lain atau sistem dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan yang tidak dapat dipisahkan.

### 3) Lingkup Luar Sistem (*enviromtment*)

Bentuk apapun yang ada diluar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut disebut lingkungan luar sistem. Lingkungan luar sistem ini dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut.

### 4) Penghubung Sistem (*Interface*)

Media yang menghubungkan sistem dengan subsistem lain disebut penghubung sistem atau *interface*. Penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsitem ke subsistem lain. Bentuk keluaran dari satu subsistem akan menjadi masukan untuk subsistem lain melalui penghubung tersebut. Dengan demikian, dapat terjadi suatu integritas sistem yang membentuk satu kesatuan.

### 5) Masukan Sistem (*Input*)

Energi yang dimasukkan ke dalam sistem disebut masukan sistem, yang dapat berupa pemeliharaan (*maintenance input*) dan sinyal (*signal input*). Contohnya, di dalam suatu unit sistem komputer, "program" adalah *maintenance input* yang digunakan untuk mengoperasikan komputernya dan "data" adalah *signal input* untuk diolah menjadi informasi.

### 6) Keluaran Sistem (*Output*)

Hasil energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran ini merupakan masukan bagi subsistem yang lain seperti sistem informasi. Keluaran yang dihasilkan adalah informasi. Informasi ini

dapat digunakan sebagai masukan untuk pengambilan keputusan atau hal-hal lain yang menjadi *input* bagi sub sistem lain.

#### 7) Pengolah Sistem (Proses)

Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran, contohnya sistem akuntansi. Sistem ini akan mengolah data transaksi menjadi laporan-laporan yang dibutuhkan oleh pihak manajemen.

#### 8) Sasaran Sistem (*Objective*)

Suatu sistem memiliki tujuan dan sasaran yang pasti dan bersifat *deterministic*. Kalau suatu sistem tidak memiliki sasaran maka operasi sistem tidak ada gunanya. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuan yang telah direncanakan.

##### a. Klasifikasi Sistem

Menurut Hutahaean (2015), “Sistem merupakan suatu bentuk integrasi antara satu komponen dan komponen lain karena sistem memiliki sasaran yang berbeda untuk setiap kasus yang terjadi didalam sistem tersebut”. Oleh karena itu sistem dapat diklasifikasikan dari beberapa sudut pandang. Adapun klasifikasi sistem diuraikan sebagai berikut:

##### 1) Sistem abstrak (*abstract system*) dan sistem fisik (*physical system*)

Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik, misalnya sistem telogi. Sedangkan sistem fisik diartikan sebagai sistem yang nampak secara fisik, misalnya sistem komputer.

##### 2) Sistem Alamiah (*natural system*) dan sistem buatan (*human made system*)

Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat manusia, misalnya sistem tata surya dan sistem reproduksi. Sedangkan sistem buatan manusia merupakan sebuah sistem yang telah dikembangkan oleh manusia itu sendiri, sistem ini bisa dirubah sesuai dengan perkembangan zaman dan kebutuhan hidup. Sistem buatan manusia secara umum bisa disesuaikan dengan kebutuhan, jika kebutuhannya berubah maka sistem yang sudah ada tadi juga bisa diberubah.

3) Sistem tertentu (*deterministic system*) dan sistem tidak tentu probablistik (*probabilistic system*)

Sistem tertentu beroperasi dengan tingkah laku yang sudah dapat diprediksi. Interaksi bagian-bagiannya dapat dideteksi dengan pasti sehingga keluaran dari sistem dapat diramalkan, misalnya sistem komputer, adalah contoh sistem yang tingkah lakunya dapat dipastikan berdasarkan program-program komputer yang dijalankan. Sedangkan sistem tidak tentu adalah sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilitas, misalnya sistem manusia.

4) Sistem terbuka (*open system*) dan sistem tertutup (*closed system*)

Sistem terbuka merupakan sistem yang berhubungan dan terpengaruh dengan lingkungan luarnya. Sedangkan sistem tertutup merupakan sistem yang tidak berhubungan dan tidak terpengaruh dengan lingkungan luarnya. Sistem ini bekerja secara otomatis tanpa adanya campur tangan dari pihak luar.

### 2.3.2 Klasifikasi Informasi

Berikut adalah pengertian informasi menurut para ahli :

a. Sutarman (2012)

Informasi adalah sekumpulan fakta (data) yang diorganisasikan dengan cara tertentu sehingga mereka mempunyai arti bagi si penerima.

b. Hartono (2013)

Informasi adalah hasil kegiatan pengolahan data yang memberikan bentuk yang lebih berarti dari suatu kejadian.

Adapun kualitas dari suatu informasi di tentukan oleh

karakteristik-karakteristik sebagai berikut :

1) Kualitas informasi (*Information quality*)

a) Akurat (*output*)

Akurat Berarti informasi harus bebas dari kesalahan-kesalahan, tidak tidak bisa atau tidak menyesatkan dan harus jelas mencerminkan maksudnya.

b) Tepat waktu (*Times lines*)

Diartikan bahwa informasi yang di gunakan bukan merupakan informasi yang telah usang dan tidak bernilai lagi, serta harus sampai pada penerima tidak terlambat sehingga membuat keputusan secara tepat waktu.

c) Relevan (*Relevant*)

Diartikan bahwa informasi yang di sampaikan mempunyai hubungan dengan masalah yang akan digunakan bersama informasi tersebut.

## 2) Akseibilitas informasi (*Information accessibility*)

### a) Ketersediaan (*Availability*)

Memberikan informasi kepada yang membutuhkan. Informasi dapat diakses oleh yang membutuhkan.

### b) Keabsahan (*admissibility*)

Keabsahan (boleh atau tidak boleh dipakai) informasi tergantung pada hukum, peraturan atau budaya pada saat tertentu.

## 3) Presentasi informasi (*Information presentation*)

### a) Tingkatan (*level of summarization*)

Perbandingan antara data asli dengan yang ditampilkan. Manipulasi data hingga tingkatan yang sesuai, semakin sederhana semakin baik

### b) Format Bentuk dimana informasi ditampilkan ke user.

Manipulasi data ke dalam bentuk yang sesuai.

## 4) Keamanan informasi (*information security*)

### a) Batasan akses (*access restriction*)

Prosedur dan teknik mengontrol user yang boleh atau tidak mengakses data pada situasi tertentu. Penggunaan *password* atau teknik lain untuk mencegah *user* yang tidak berhak.

### b) Enkripsi (*encryption*)



Konversi data ke bentuk tertentu sehingga tidak dapat dibaca oleh *user* yang tidak berhak.

### **2.3.3 Nilai informasi**

Nilai dari informasi (*value of information*) ditentukan dari dua hal, yaitu manfaat dan biaya mendapatkannya. Suatu informasi dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya mendapatkannya. Sebagian besar informasi tidak dapat persis ditaksir keuntungannya dengan satuan nilai uang, tetapi dapat ditaksir nilai efektivitasnya. Pengukuran nilai informasi biasanya dihubungkan dengan analisis *cost effectiveness* atau *cost benefit*.

### **2.3.4 Komponen Sistem Informasi**

Menurut Rusdiana dan Irfan (2014), sistem informasi terdiri dari komponen-komponen yang disebut blok bangunan yang saling berinteraksi antara satu dengan yang lain dan membentuk suatu kesatuan untuk mencapai sasarannya.

#### **a. Blok Masukan**

Input mewakili data yang masuk ke dalam sistem informasi. Input disini termasuk metode-metode dan media untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen-dokumen dasar.

#### **b. Blok Model**

Blok ini terdiri dari kombinasi prosedur, logika dan model matematik yang akan memanipulasi data input dan data yang tersimpan di basis data

dengan cara yang sudah tertentu untuk menghasilkan keluaran yang diinginkan.

c. Blok Keluaran

Produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

d. Blok Teknologi

Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian dari sistem secara keseluruhan. Teknologi terdiri dari 3 bagian utama, yaitu teknisi (*humanware* atau *brainware*), perangkat lunak (*software*), dan perangkat keras (*hardware*).

e. Blok Basis Data

Basis data (*database*) merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya. Data perlu disimpan di dalam basis data untuk keperluan penyediaan informasi lebih lanjut.

f. Blok Kendali

Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah

ataupun bila terlanjur terjadi kesalahan-kesalahan dapat langsung cepat diatasi.

Pengelompokan komponen-komponen sistem informasi berbasis komputer adalah sebagai berikut:

1) Perangkat keras (*hardware*)

*Hardware* ini merupakan peralatan fisik yang dapat digunakan untuk mengumpulkan, memasukkan, memproses, menyimpan, dan mengeluarkan hasil pengolahan data dalam bentuk informasi. Perangkat lunak (*software*) *Software* adalah kumpulan dari program-program yang digunakan untuk menjalankan aplikasi tertentu pada komputer.

2) Manusia (*brainware*)

*Brainware* dalam sistem informasi berperan sebagai pemberi dan pengguna informasi.

3) Prosedur (*procedure*)

Prosedur adalah rangkaian aktivitas atau kegiatan yang dilakukan secara berulang-ulang dengan cara yang sama.

4) Basis data (*database*)

*Database* merupakan kumpulan data-data yang tersimpan di dalam media penyimpanan di suatu perusahaan (arti luas) atau di dalam komputer (arti sempit).

5) Jaringan komunikasi (*communication network*)

Jaringan telekomunikasi saat ini menghubungkan beberapa daratan dan lautan untuk memindahkan data dalam jumlah besar.

### **2.3.5 Sistem Informasi Integrasi**

Sistem integrasi (integrated system) merupakan sebuah rangkaian proses untuk menghubungkan beberapa sistem komputerisasi dan software aplikasi, baik secara fisik maupun secara fungsional. Sistem terintegrasi akan menggabungkan komponen sub-sub sistem ke dalam satu sistem dan menjamin fungsi-fungsi dari sub sistem tersebut sebagai satu kesatuan system. Didasarkan pada beberapa penelitian mengenai sistem informasi yang diintegrasikan di kawasan universitas, ada beberapa hal penting yang dapat disimpulkan mengenai kepentingan terwujudnya implementasi sistem ini di berbagai universitas. Banyaknya informasi yang menyangkut mengenai segala komponen dalam universitas mengharuskan kecepatan dalam pengolahan data yang bersangkutan. Melalui penggunaan sistem informasi, berbagai informasi tersebut dapat diakses oleh berbagai pihak universitas untuk mempermudah kelancaran dalam menjalankan aktivitas – aktivitas dalam universitas. Sistem informasi integrasi dapat bersifat hirarkis yaitu pada tingkat transaksi akan memberikan masukan data kepada sistem tingkat manajerial atau sering pula dalam arah sebaliknya. Interaksi hirarkis adalah paling banyak diidentifikasi dan diintegrasikan karena manajer mengetahui bahwa informasi harus diringkaskan menurut jalur hirarki disamping sistem yang bersangkutan ada di bawah satu garis komando

dan karena manajer dalam bidang fungsional akan lebih banyak mengetahui data apa yang ada dalam sistemnya.

## **2.4 Pelayanan Pengaduan**

Pelayanan dapat di artikan menurut beberapa ahli. Menurut (KBBI) adalah sebagai suatu usaha untuk membantu menyiapkan atau mengurus apa yang diperlukan orang lain. pelayanan adalah kegiatan yang dilakukan oleh seseorang atau sekelompok orang dengan landas faktor materi melalui sistem, prosedur dan metode tertentu dalam rangka usaha memenuhi kepentingan sesuai dengan haknya. Pelayanan hakikatnya adalah serangkaian kegiatan, karena itu pelayanan merupakan sebuah proses. Sebagai proses, pelayanan berlangsung secara rutin dan berkesinambungan, meliputi seluruh kehidupan orang dalam masyarakat. Pengaduan, adalah laporan yang mengandung informasi atau indikasi terjadinya penyalahgunaan wewenang, penyimpangan atau pelanggaran perilaku yang dilakukan oleh aparat pengadilan, yang berasal dari masyarakat, anggota, instansi peradilan, instansi diluar pengadilan, maupun dari media massa dan sumber-sumber informasi lain yang relevan. Pengaduan, adalah laporan yang mengandung informasi atau indikasi terjadinya penyalahgunaan wewenang, penyimpangan atau pelanggaran perilaku yang dilakukan oleh aparat pengadilan, yang berasal dari masyarakat, anggota, instansi peradilan, instansi di luar pengadilan, maupun dari media massa dan sumber-sumber informasi lain yang relevan.

## 2.5 *Flowchart* (Bagan Alir)

Bagan alir (*flowchart*) Menurut Romney & Steinbart (2012) bagan alir (*flowchart*) adalah suatu teknik analitis yang digunakan untuk menggambarkan beberapa aspek pada system informasi ke dalam suatu cara yang jelas, ringkas, dan logis.

Simbol *flowchart* menurut Romney & Steinbart (2012), dibagi ke dalam empat kategori berikut:

- a. *Input/output symbols*, merupakan perangkat atau media yang menyediakan input atau mencatat output dari proses operasi.
- b. *Processing symbols*, menunjukkan tipe perangkat apa yang digunakan untuk memproses data atau mengindikasikan kapan sebuah proses dilakukan secara manual.
- c. *Storage symbols*, menunjukkan perangkat yang digunakan untuk menyimpan data yang tidak sedang digunakan oleh sistem.
- d. *Flow and miscellaneous symbols*, mengindikasikan aliran data dan barang, serta mewakili operasi dimana flowchart diawali atau diakhiri, dimana keputusan dibuat, dan kapan memberikan penjelasan tambahan pada *flowchart*.

### 2.5.1 Jenis-Jenis *Flowchart*

*Flowchart* terbagi atas lima jenis, yaitu:

#### a. *Flowchart* sistem

*Flowchart* Sistem merupakan bagan yang menunjukkan alur kerja atau apa yang sedang dikerjakan di dalam sistem secara keseluruhan dan menjelaskan urutan dari prosedur-prosedur yang ada di dalam sistem. Dengan kata lain, *flowchart* ini merupakan deskripsi secara grafik dari urutan prosedur-prosedur yang terkombinasi yang membentuk suatu sistem.

*Flowchart* Sistem terdiri dari data yang mengalir melalui system dan proses yang mentransformasikan data itu. Data dan proses dalam *flowchart* sistem dapat digambarkan secara *online* (dihubungkan langsung dengan komputer) atau *offline* (tidak dihubungkan langsung dengan komputer, misalnya mesin tik, *cash register* atau kalkulator).

#### b. *Flowchart* Dokumen

*Flowchart Paperwork* menelusuri alur dari data yang ditulis melalui sistem. *Flowchart Paperwork* sering disebut juga dengan *Flowchart* Dokumen. Kegunaan utamanya adalah untuk menelusuri alur form dan laporan sistem dari satu bagian ke bagian lain baik bagaimana alur form dan laporan diproses, dicatat dan disimpan.

#### c. *Flowchart* Skematik

*Flowchart* Skematik mirip dengan *Flowchart* Sistem yang menggambarkan suatu sistem atau prosedur. *Flowchart* Skematik ini bukan hanya menggunakan simbol-simbol *flowchart* standar, tetapi juga

menggunakan gambar-gambar komputer, *peripheral*, form-form atau peralatan lain yang digunakan dalam sistem.

*Flowchart* Skematik digunakan sebagai alat komunikasi antara analis sistem dengan seseorang yang tidak familiar dengan simbol-simbol *flowchart* yang konvensional. Pemakaian gambar sebagai ganti dari simbol-simbol *flowchart* akan menghemat waktu yang dibutuhkan oleh seseorang untuk mempelajari simbol abstrak sebelum dapat mengerti *flowchart*.

Gambar-gambar ini mengurangi kemungkinan salah pengertian tentang sistem, hal ini disebabkan oleh ketidak-mengertian tentang simbol-simbol yang digunakan. Gambar-gambar juga memudahkan pengamat untuk mengerti segala sesuatu yang dimaksudkan oleh analis, sehingga hasilnya lebih menyenangkan dan tanpa ada salah pengertian.

#### d. *Flowchart* Program

*Flowchart* Program dihasilkan dari *Flowchart* Sistem. *Flowchart* Program merupakan keterangan yang lebih rinci tentang bagaimana setiap langkah program atau prosedur sesungguhnya dilaksanakan. *Flowchart* ini menunjukkan setiap langkah program atau prosedur dalam urutan yang tepat saat terjadi.

*Programmer* menggunakan *flowchart* program untuk menggambarkan urutan instruksi dari program komputer. Analis Sistem menggunakan *flowchart* program untuk menggambarkan urutan tugas-tugas pekerjaan dalam suatu prosedur atau operasi.

#### e. *Flowchart* Proses



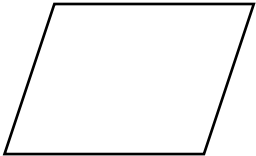
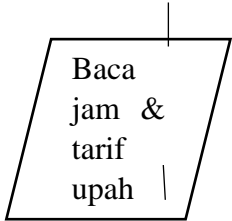

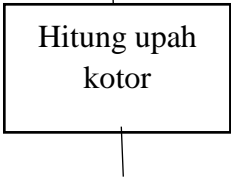
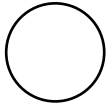
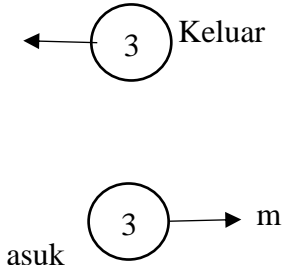

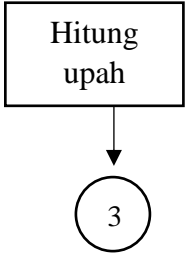
*Flowchart* Proses merupakan teknik penggambaran rekayasa industrial yang memecah dan menganalisis langkah-langkah selanjutnya dalam suatu prosedur atau sistem.

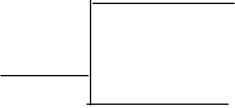
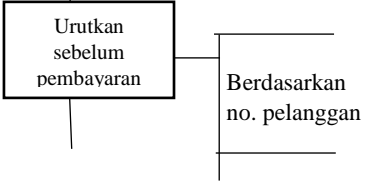
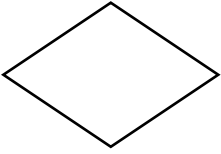
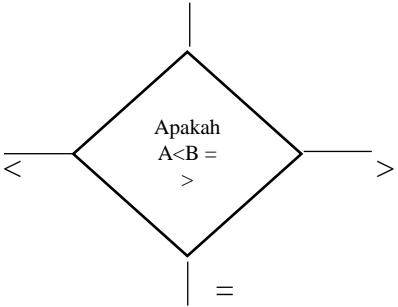

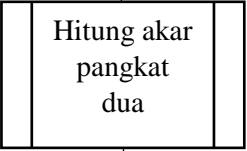
*Flowchart* Proses digunakan oleh perekayasa industrial dalam mempelajari dan mengembangkan proses-proses *manufacturing*. Dalam analisis sistem, *flowchart* ini digunakan secara efektif untuk menelusuri alur suatu laporan atau form.


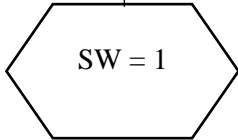

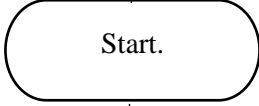
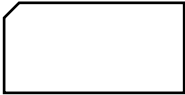
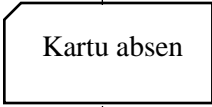

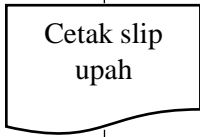
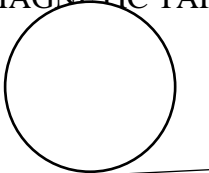
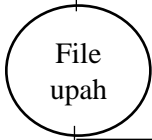
### **2.5.2 Simbol-Simbol *Flowchart***

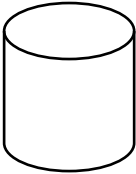

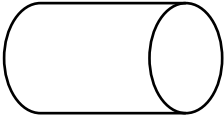
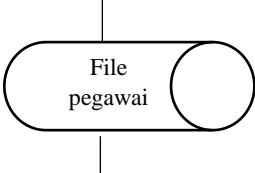
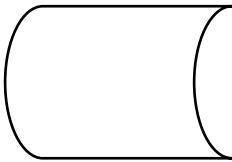
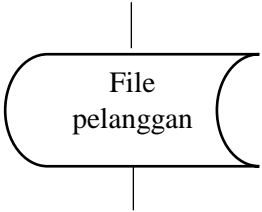
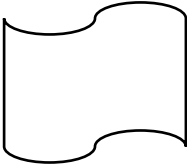


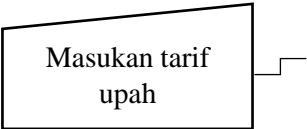
Simbol-simbol *flowchart* yang biasanya dipakai adalah simbol-simbol standar yang dikeluarkan oleh ANSI dan ISO.


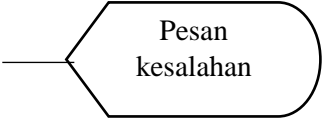
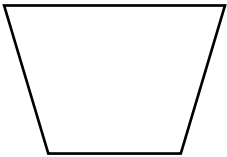
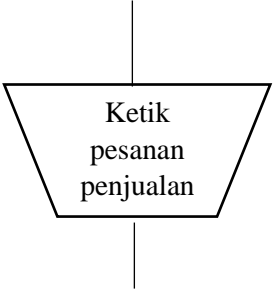
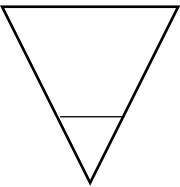
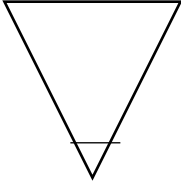
**Tabel 2.1** simbol *Flowchart*

SIMBOL	ARTI	CONTOH
<p>INPUT</p> 	Merepresentasikan input data atau output data yang diproses atau informasi	
<p>PROSES</p> 	Mempresentasikan operasi	
<p>PENGHUBUNG</p> 	Keluar ke atau masuk dari bagian lain flowchart khususnya halaman yang sama	
<p>ANAK PANAHA</p> 	Merepresentasikan alur kerja	

<p>PENJELASAN</p> 	<p>Digunakan untuk komentar tambahan</p>	
<p>KEPUTUSAN</p> 	<p>Keputusan dalam program.</p>	
<p>PREDEFINED PROCESS</p> 	<p>Rincian operasi berada di tempat lain.</p>	

SIMBOL	ARTI	CONTOH
<b>PREPARATION</b> 	Pemberian harga awal.	
<b>TERMINAL POINTS</b> 	Awal/akhir flowchart.	
<b>PUNCHED CARD</b> 	Input/output yang menggunakan kartu berlubang	
<b>DOKUMEN</b> 	I/O dalam format yang dicetak.	
<b>MAGNETIC TAPE</b> 	I/O yang menggunakan pita magnetik.	

SIMBOL	ARTI	CONTOH
<p>MAGNETIC DISK</p> 	I/O yang menggunakan disk magnetik.	
<p>MAGNETIC DRUM</p> 	I/O yang menggunakan drum magnetik	
<p>ONLINE STORAGE</p> 	I/O yang menggunakan penyimpanan akses langsung.	
<p>PUNCHED TAPE</p> 	I/O yang menggunakan pita kertas berlubang.	
<p>MANUAL INPUT</p> 	Input yang dimasukan secara manual dari keyboard.	

SIMBOL	ARTI	CONTOH
DISPLAY 	Output yang ditampilkan pada terminal.	
MANUAL OPERATION 	Operasi manual.	
OFF-LINE STORAGE 	Penyimpanan yang tidak dapat diakses oleh komputer secara langsung.	

## 2.6 Pengertian UML ( *Unified Modeling Language* )

### A. Pengertian UML

Menurut Rosa Dan Shalahuddin (2015:133) “UML (Unified Modelling Language) adalah salah satu standar bahasa yang banyak digunakan didunia industri untuk mendefinisikan requirement, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasikan objek.

Sedangkan menurut Ariani R. Sukamto dalam Taufik (2017) “UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram teks-teks pendukung”.

Pendapat lainnya menurut Fowler, M. dalam (B. O. Lubis, 2016) UML (Unified Modeling Language) adalah “Keluarga notasi grafis yang didukung

oleh meta-model tunggal, yang membantu pendeskripsian dan desain sistem perangkat lunak, khususnya sistem yang dibangun menggunakan pemrograman berorientansi objek (OO). Definisi ini merupakan definisi yang sederhana”.

Jadi UML (Unified Modeling Language) dapat diartikan sebagai bahasa visual untuk menggambarkan definisi-definisi tentang requirement, membuat analisis dan desain serta menggambar arsitektur dalam pemrograman berorientasikan objek dengan menggunakan teks-teks pendukung.

#### B. Langkah-langkah penggunaan UML

Menurut Adi Nugroho dalam Esa Wijayanti (2014:15), langkah-langkah penggunaan Unified Modeling Language (UML) sebagai berikut:

1. Buatlah daftar business process dari level tertinggi untuk mendefinisikan aktifitas dan proses yang mungkin muncul
2. Petakan use case untuk setiap business process untuk mendefinisikan dengan tepat fungsional yang harus disediakan oleh sistem, kemudian perhalus use case diagram dan lengkapi dengan requirement, constraints dan catatan-catatan lain
3. Buatlah deployment diagram secara kasar untuk mendefinisikan arsitektur fisik sistem
4. Berdasarkan use case diagram, mulailah membuat activity diagram
5. Definisikan obyek-obyek level atas package atau domain dan buatlah sequence dan/atau collaboration untuk tiap alur pekerjaan, jika sebuah use case memiliki kemungkinan alur normal dan error, buat lagi satu diagram untuk masing-masing alur

6. Buatlah rancangan user interface model yang menyediakan antar muka bagi pengguna untuk menjalankan skenario use case
7. Berdasarkan model-model yang sudah ada, buatlah class diagram. Setiap package atau domain dipecah menjadi hirarki class lengkap dengan atribut dan metodenya. Akan lebih baik jika untuk setiap class dibuat unit test untuk menguji fungsionalitas class dan interaksi dengan class lain
8. Setelah class diagram dibuat, kita dapat melihat kemungkinan pengelompokan class menjadi komponen-komponen karena itu buatlah component diagram pada tahap ini. Juga definisikan test integrasi untuk setiap komponen meyakinkan ia bereaksi dengan baik
9. Perhalus deployment diagram yang sudah dibuat. Detilkan kemampuan dan requirement piranti lunak, sistem operasi, jaringan dan sebagainya. Petakan komponen kedalam node
10. Mulailah membangun sistem, ada dua pendekatan yang tepat digunakan
  - a. Pendekatan use case dengan menetapkan setiap use case kepada tim pengembang tertentu untuk mengembangkan unit kode yang lengkap dengan test.
  - b. Pendekatan komponen yaitu menetapkan setiap komponen kepada tim pengembang tertentu.



### C. Model UML

Menurut Widodo (2011:10), beberapa *literature* menyebutkan bahwa UML menyediakan sembilan jenis diagram, yang lain menyebutkan delapan karena ada beberapa diagram yang digabung, misalnya diagram komunikasi, diagram urutan dan diagram pewaktuan digabung menjadi diagram interaksi. Namun demikian model-model itu dapat dikelompokkan berdasarkan sifatnya yaitu statis atau dinamis. Beberapa jenis diagram dalam UML yaitu :

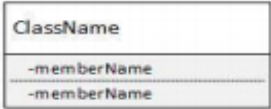




#### 2.6.1 Diagram Kelas (*Class Diagram*)

*Class diagram* merupakan salah satu diagram utama dari UML untuk menggambarkan class atau blueprint object pada sebuah sistem. Analisis pembentukan class diagram merupakan aktivitas inti yang sangat mempengaruhi arsitektur piranti lunak yang dirancang hingga ke tahap pengkodean.

Menurut Yuni Sugiarti (2013), “*Diagram kelas menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem*”. Kelas memiliki apa yang disebut atribut dan method atau operasi. Berikut penjelasan atribut dan method :

- a. Atribut merupakan *variable-variable* yang dimiliki oleh suatu kelas
- b. Operasi atau *method* adalah fungsi-fungsi yang dimiliki oleh suatu kelas

Tabel 2.2 : *Class Diagram Symbol*

No.	Simbol	Deskripsi
1.	Kelas 	Kelas pada struktur sistem
2.	Antarmuka/ <i>interface</i> 	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek
3.	Asosiasi/ <i>association</i> 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
4.	Asosiasi berarah/ <i>directed association</i> 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
5.	Generalisasi 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum-khusus)

Hubungan antar *class* pada diagram juga digambarkan bagaimana interaksi hubungan antar class dalam sebuah konstruksi piranti lunak seperti hubungan asosiasi, agregasi, komposisi, *inheritance* dan *generalization*.

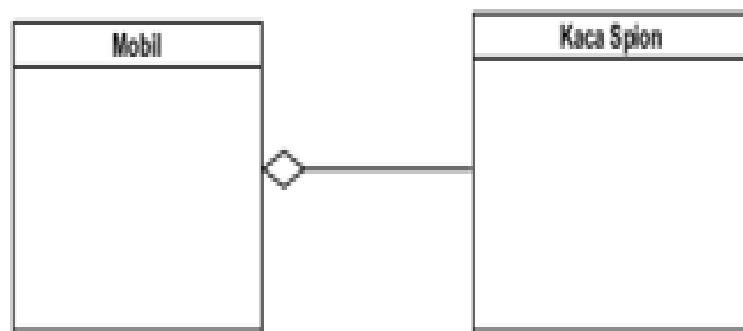
a. Asosiasi

Yaitu hubungan statis antar class. Umumnya menggambarkan *class* yang memiliki atribut berupa *class* lain, atau *class* yang harus mengetahui eksistensi class lain.

b. Agregasi

Yaitu hubungan antar class yang menyatakan hubungan has-a. Contohnya : Mobil dengan kaca spion.

**Gambar 2.1 : Mobil dengan Kaca Spion**



Sumber : John W. Satzinger, R. B. (2012)

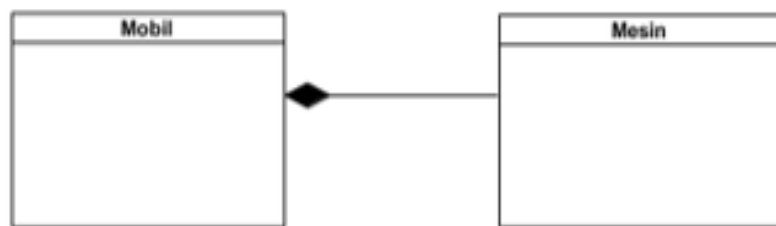
Bila kita memodelkan sebuah mobil, maka dapat dikatakan bahwa mobil mempunyai kaca spion. Secara logis dapat dikatakan sebuah mobil dapat berfungsi walaupun tanpa kaca spion. Mobil dapat dipisahkan dari kaca spion.

c. Komposisi

Komposisi yaitu hubungan antar *class* yang menyatakan *part-of*.

Contohnya : Mobil dengan mesin

**Gambar 2.2** : Mobil dengan Mesin



Sumber :John W. Satzinger, R. B. (2012)

Bila kita memodelkan sebuah mobil, maka dapat dikatakan bahwa mesin adalah bagian dari mobil. Secara logis dapat dikatakan bahwa mobil tidak dapat dipisahkan dari mesin. Bila mesin tidak ada di sebuah mobil maka mobil itu tidak berfungsi sebagai mobil.

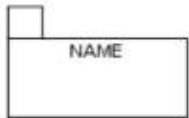
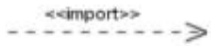
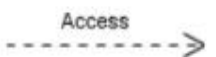
*d. Inheritance dan Generalization*

*Inheritance* adalah hubungan hirarkis antar *class*. *Class* dapat diturunkan dari class lain dan mewarisi semua atribut dan metoda class asalnya dan menambahkan fungsionalitas baru, sehingga ia disebut anak dari class yang diwarisinya. Kebalikan dari pewarisan adalah generalisasi.

### 2.6.2 Diagram Paket (*Package Diagram*)

Diagram ini memperlihatkan kumpulan kelas-kelas, merupakan bagian dari diagram komponen. Diagram paket bersifat statis.

**Tabel 2.3 : Package Diagram Symbol**

Contract	Deskripsi	Lambang
Package	Sekelompok elemen-elemen model	
Import	Suatu dependency yang mengindikasikan isi tujuan paket secara umum yang ditambahkan kedalam sumber paket	
Access	Suatu dependency yang mengindikasikan isi tujuan paket secara umum yang bisa digunakan pada nama sumber paket	

### 2.6.3 Use Case Diagram

Menurut Rosa dan Shalahuddin (2015: 155) Use case atau diagram use case merupakan pemodelan untuk kelakuan (behavior) sistem informasi yang akan dibuat.


Jadi dapat disimpulkan bahwa *use case* diagram adalah diagram yang merupakan representasi visual yang mewakili interaksi antara pengguna dan sistem untuk menunjukkan peran dari pengguna dan bagaimana peran-peran menggunakan sistem.







*Use case* diagram digunakan untuk memodelkan bisnis proses berdasarkan perspektif pengguna sistem. *Use case* diagram terdiri atas diagram untuk *use case* dan *actor*. “*Actor* mempresentasikan orang yang akan mengoperasikan atau berinteraksi dengan sistem aplikasi” (Rosa A.S dan M. Shalahuddin, 2014:150).




*Use case* mempresentasikan operasi-operasi yang dilakukan oleh *actor*. *Use case* digambarkan berbentuk *elips* dengan nama operasi dituliskan di dalamnya. *Actor* melakukan operasi diubungkan dengan garis lurus ke *use case*. Manfaat dari *use case* adalah :

- a. Untuk memudahkan komunikasi dengan menggunakan *domain expert* dan juga *end user*
- b. Adanya *Interface* yang harus dimiliki oleh sebuah sistem
- c. Memberikan kepastian pemahaman yang pas, tentang *requirement* atau juga kebutuhan sebuah sistem
- d. Dapat digunakan untuk mengidentifikasi, siapa yang sedang berinteraksi dengan sistem, dan juga harus dilakukan untuk sistem tersebut
- e. Digunakan untuk verifikasi

**Tabel 2.4 : Use Case Diagram Symbol**

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .

2		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri ( <i>independent</i> ) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri ( <i>independent</i> ).
3		<i>Generalization</i>	Hubungan dimana objek anak ( <i>descendent</i> ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancestor</i> ).
4		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .
5		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.

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

Sumber: Buku Karangan Rosa A.S dan M. Shalahuddin (2014:156)

#### 2.6.4 Diagram Interaksi dan *Sequence* (*Sequence Diagram*)

Menurut Vidia (2013:20), ”*Sequence diagram* dibuat berdasarkan *activity diagram* dan *class diagram*. *Sequence diagram* menggambarkan aliran pesan yang terjadi antar kelas yang dideskripsikan pada *class diagram* dengan menggunakan operasi yang dimiliki kelas tersebut. Untuk aliran pesan, *sequence diagram* merujuk pada alur sistem *activity diagram* yang telah dibuat sebelumnya”.

Menurut Wijayanto (2013:35), ”*Sequence diagram* dibuat berdasarkan *activity diagram* dan *class diagram* yang telah dibuat, maka digambarkan *sequence diagram* yang menggambarkan aliran pesan yang terjadi antar kelas dengan menggunakan operasi yang dimiliki kelas tersebut”.



Sequence diagram bisa digunakan untuk menjelaskan sebuah serangkaian langkah-langkah mengirimkan pesan antar satu *lifeline* ke *lifeline* yang lain. Setiap pesan yang dikirimkan bisa memberikan respon (*return*) relatif pada skenario yang dirancang di *use case* diagram. Interaksi yang terjadi bisa bersifat instansiasi sebuah object maupun static method dari sebuah class.

Komponen-komponen *sequence* diagram adalah :

a. Object

Adalah komponen berbentuk kotak yang mewakili sebuah class atau object. Mereka mendemonstrasikan bagaimana sebuah object berperilaku pada sebuah sistem

b. *Activation Boxes*

Adalah komponen yang berbentuk persegi panjang yang menggambarkan waktu yang diperlukan sebuah object untuk menyelesaikan tugas. Lebih lama waktu yang diperlukan, maka *activation boxes* akan lebih panjang

c. *Actors*



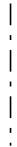
Adalah komponen yang berbentuk *stick figure*. Komponen yang mewakili seorang pengguna yang berinteraksi dengan sistem



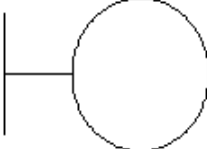
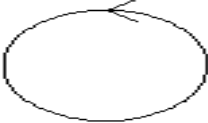
d. *Lifeline*


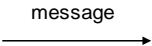

Merupakan komponen berbentuk garis putus-putus. *Lifeline* biasanya memuat kotak yang berisi nama dari sebuah object. Berfungsi menggambarkan aktifitas dari object

Berikut simbol sequence diagram yang dapat dilihat pada tabel berikut :

**Tabel 2.5 : Sequence Diagram Symbol**

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	<i>Actor</i> juga dapat berkomunikasi dengan <i>object</i> , maka <i>actor</i> juga dapat diurutkan sebagai kolom.
2		<i>Object</i>	<i>Object</i> atau biasa juga disebut partisipan merupakan instance dari sebuah class dan dituliskan tersusun secara horizontal.  Digambarkan sebagai sebuah class (kotak) dengan nama objek didalamnya yang diawali dengan sebuah titik koma.
3		<i>Lifeline</i>	<i>Lifeline</i> mengindikasikan keberadaan sebuah object dalam basis waktu. Notasi untuk <i>Lifeline</i> adalah garis putus-putus vertikal yang ditarik dari sebuah objek.

4		<i>Message Return</i>	Menggambarkan pesan/hubungan antar obyek, yang menunjukkan urutan kejadian yang terjadi.
5		<i>Activation</i>	<i>Activation</i> dinotasikan sebagai sebuah kotak segi empat yang digambar pada sebuah <i>lifeline</i> . <i>Activation</i> mengindikasikan sebuah objek yang melakukan suatu aksi
6		<i>Boundary</i>	Boundary terletak di antara sistem dengan dunia sekelilingnya. Semua form, laporan-laporan, antar muka ke perangkat keras seperti printer atau scanner dan antar muka ke sistem lainnya adalah termasuk dalam kategori.
7		<i>Control</i>	<i>Control</i> berhubungan dengan fungsionalitas seperti pemanfaatan sumber daya, pemrosesan terdistribusi, atau penanganan kesalahan.

8		<i>Entity</i>	<i>Entity</i> digunakan menangani informasi yang mungkin akan disimpan secara permanen. <i>Entity</i> bisa juga merupakan sebuah tabel pada struktur basis data.
9		<i>Message</i>	<i>Message</i> , digambarkan dengan anak panah horizontal antara <i>Activation</i> . <i>Message</i> mengindikasikan komunikasi antara <i>object-object</i> .
10		<i>Self-Message</i>	<i>Self-message</i> atau panggilan mandiri mengindikasikan komunikasi kembali kedalam sebuah objek itu sendiri.

Sumber :John W. Satzinger, R. B. (2012)

### 2.6.5 Activity Diagram




Menurut Martin Fowler (2015:163) “*Activity diagram adalah teknik untuk menggambarkan logika prosedural, proses bisnis, dan jalur kerja*”. Dalam beberapa hal, activity diagram memainkan peran mirip diagram alir, tetapi prinsip antara notasi diagram alir adalah *activity* diagram mendukung behavior paralel. Node pada sebuah activity diagram disebut action, sehingga diagram tersebut menampilkan sebuah *activity* yang tersusun dari action.





Sesuai dengan namanya diagram ini menggambarkan tentang aktifitas yang terjadi pada sistem. Dari pertama sampai akhir, diagram ini menunjukkan langkah-langkah dalam proses kerja sistem yang kita buat. Sebagai contoh langkah-langkah memasak nasi, tetapi kita akan menjelaskannya dalam bentuk grafik. Struktur diagram ini juga mirip dengan *flowchart*.

Adapun fungsi dari *activity* diagram yaitu sebagai berikut :

- a. Menggambarkan proses bisnis dan urutan aktifitas dalam sebuah proses.
- b. Memperlihatkan urutan aktifitas proses pada sistem.
- c. *Activity* diagram dibuat berdasarkan sebuah atau beberapa *use case* pada *use case* diagram.

**Tabel 2.6 : Activity Diagram Symbol**

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.

4		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
5		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran
6		<i>Join Node</i>	Banyak aliran yang pada tahap tertentu berubah menjadi beberapa satu aliran
7		Decision	Pilihan untuk mengambil keputusan

Sumber : John W. Satzinger, R. B. (2012)

### 2.6.6 Flowchart

Menurut Wibawanto (2017:20) “Flowchart adalah suatu bagan dengan simbol-simbol tertentu yang menggambarkan urutan proses secara mendetail dan hubungan antara suatu proses (intruksi) dengan proses lainnya dalam suatu program”. Diagram alur dapat menunjukkan secara jelas, arus pengendalian suatu algoritma yakni bagaimana melaksanakan suatu rangkaian kegiatan secara logis dan sistematis

Menurut pandangan Indrajani (2011) “*flowchart merupakan gambaran secara grafik dari langkah-langkah dan urutan prosedur suatu program*”.

Dari pendapat para ahli diatas bisa disimpulkan bahwa *flowchart* adalah diagram alir dari algoritma-algoritma dan gambaran secara grafik dalam suatu program, yang menunjukkan prosedur system secara logika.

*Flowchart* sangat erat kaitannya dengan Data Flow Diagram (DFD). Menurut Rosa A.S M. Shalahudin (2016), Data Flow Diagram (DFD) atau dalam bahasa Indonesia menjadi Diagram Alir Data (DAD) adalah representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (input) dan keluaran (Output).

### **2.6.7 Jenis-jenis *Flowchart***

*Jenis – jenis Flowchart* menurut Menurut Jogiyanto (2012) dalam buku Analisis dan Disain Sistem Informasi , jenis-jenis flowchart terdiri dari :

#### **a. Bagan Alir Sistem (*SystemFlowchart*)**

Bagan alir sistem merupakan bagan yang menunjukkan arus pekerjaan secara keseluruhan dari sistem. Bagan ini menjelaskan urutan dari prosedur-prosedur yang ada di dalam sistem.

#### **b. Bagan Alir Dokumen ( *Document Flowchart* )**

Bagan alir dokumen merupakan bagan alir yang menunjukkan arus dari laporan dan formulir termasuk tembusan-tembusannya.

#### **c. Bagan Alir Skematik (*Schematic Flowchart*)**

Bagan alir skematik merupakan bagan alir yang mirip dengan bagan alir sistem, yaitu untuk menggambarkan prosedur di dalam sistem.

#### **d. Bagan Alir Program (*Program Flowchart*)**












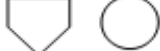
Bagan alir program merupakan bagan yang menjelaskan secara rinci langkah-langkah dari proses program.

e. Bagan Alir Proses (*Process Flowchart* )

Bagan alir proses merupakan bagan alir yang banyak digunakan di teknik industri. Bagan alir ini juga berguna bagi analisis sistem untuk menggambarkan proses dalam suatu prosedur.



**Tabel 2.7 : Simbol Bagan Alir**

No	Gambar	Nama	Keterangan
1		Dokumen	Menunjukkan dokumen <i>input</i> dan <i>output</i> baik untuk proses manual, mekanik atau komputer.
2		Proses Manual	Merupakan proses manual dalam <i>flowchart</i> .
3		Simbol Proses Komputerisasi	Menunjukkan kegiatan proses dari operasi program komputer.
4		<i>File Hardisk/ Database</i>	Menunjukkan kegiatan <i>input</i> atau <i>output</i> menggunakan <i>hardisk</i> .
5		<i>Offline Storage</i>	Menunjukkan file non-komputer yang diarsip urut tanggal (cronological)
6		Simbol Keyboard	Merupakan <i>input</i> data yang menggunakan <i>online keyboard</i> .
7		Arus dokumen/ pemrosesan	Menunjukkan arus dari proses.
8		Keputusan	Menunjukkan tahapan pembuatan keputusan
9		Terminal	Menunjukkan awal dan akhir dari bagan alir dokumen.
10		<i>Input/Output</i>	Mewakili data <i>input/output</i> .
11		Penjelasan	Menunjukkan penjelasan dari suatu proses
12		<i>Connector</i>	Menunjukkan penghubung ke halaman yang sama atau ke halaman lain
13		Arus dari Jaringan	Data melalui channel komunikasi

Sumber : Jogyanto, Analisis dan Design Sistem

## 2.7 *Enterprise Architect*

*Enterprise Architect* adalah sebuah *platform* visual untuk merancang dan membangun sebuah sistem perangkat lunak, untuk pemodelan proses bisnis, dan untuk tujuan pemodelan lebih umum.

*Enterprise Architect* adalah alat progresif yang mencakup semua aspek siklus pengembangan, memberikan keterlacakan penuh dari tahap desain awal hingga penerapan, pemeliharaan, dan kontrol perubahan.

## 2.8 *Balsamiq Mockup*

Menurut yang dilansir dalam situs website balsamiq (<https://balsamiq.com/wireframes/>) Balsamiq Wireframes atau biasa disebut Balsamiq Mockup adalah alat wireframing UI dengan kualitas rendah dan cepat yang mereproduksi pengalaman membuat sketsa pada notepad atau papan tulis, tetapi menggunakan komputer.

## 2.9 *Konsep Dasar Literature Review*

*Literature review* adalah sekumpulan jurnal, atau penelitian lain yang akan menjadi sebuah acuan penelitian peneliti, dalam penelitian yang dilakukan ilmuwan sebelumnya dapat memberikan masukan atau ide yang didalamnya hampir memiliki topik penelitian yang sama.

Menurut Deviachrista, (2013) *Literature review* adalah uraian tentang teori, temuan dan bahan penelitian lainnya yang diperoleh dari bahan acuan untuk dijadikan landasan kegiatan penelitian untuk menyusun kerangka pemikiran yang jelas dari perumusan masalah yang ingin diteliti

Berdasarkan pendapat di atas, dapat disimpulkan *literature review* adalah berupa bahan uraian teSori yang diperoleh dan digunakan untuk melihat dan menganalisa nilai tambah penelitian serta untuk dijadikan landasan kegiatan penelitian.

### **2.9.1 Manfaat *Literature Review***

Manfaat dari Studi Pustaka (*Literature Review*) yaitu:

- 1) Mengidentifikasi kesenjangan (*identify gaps*).
- 2) Menghindari pembuatan ulang (*reinventing the wheel*).
- 3) Mengidentifikasi metode yang pernah dilakukan.
- 4) Meneruskan penelitian sebelumnya, serta mengetahui orang lain yang spesialisasi dan area penelitiannya sama di bidang ini

### **2.9.2 Sumber *Literature Review***

Adapun *litteratur riview* yang penelitian ambil sebagai landasan awal ataupun pendukung dari kegiatan penelitian, yaitu:

1. Berdasarkan Penelitian Mahbub, Asep Ramdhani and Hantoro, Kusdarnowo and Suryanih, Yanih (2020) ” Analisis dan perancangan layanan pengaduan masyarakat desa berbasis web dengan pendekatan uml”
2. Berdasarkan Penelitian JUARSYAH, Ishak; MULYONO, Herry (2021). “Analisis Dan Perancangan Sistem Informasi Pengaduan

Masyarakat Berbasis Android Pada Dinas Komunikasi Dan Informatika Kota Jambi”.

3. Berdasarkan Penelitian SANSENA, Yudha (2021).”Implementasi Sistem Layanan Pengaduan Masyarakat Kecamatan Medan Amplas Berbasis Website”.