

## BAB II

### LANDASAN TEORI

#### 2.1. Konsep Dasar

##### 2.1.1. Definisi Sitem

Menurut Hartono (2016:1) “Suatu sistem adalah suatu jaringan dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu”.

Menurut Mulyadi (2016:5) “Sistem adalah suatu jaringan prosedur yang dibuat menurut pola yang terpadu untuk melaksanakan kegiatan pokok perusahaan”.

Menurut Romney dan Steinbart (2015:3) “Sistem adalah rangkaian dari dua atau lebih komponen-komponen yang saling berhubungan, yang berinteraksi untuk mencapai suatu tujuan. Sebagian besar sistem terdiri dari subsistem yang lebih kecil yang mendukung sistem yang lebih besar”

Menurut Pratama (2014:7) sistem di definisikan sebagai “sekumpulan prosedur yang saling berkalitan dan saling terhubung untuk melakukan suatu tugas bersama-sama.”

Menurut Tohari (2014:2) sistem di definisikan sebagai “kumpulan atau himpunan dari unsur atau *variabel-variabel* yang saling terkait, saling berinteraksi, dan saling tergantung satu sama lain untuk mencapai tujuan.”

### 2.1.2. Karakteristik Sistem

Menurut Hartono (2016:3) bahwa “suatu sistem mempunyai karakteristik atau sifat-sifat tertentu, yaitu memiliki komponen-komponen (*components*), batasan sistem (*boundary*), lingkungan luar sistem (*Environments*), penghubung (*Interface*), masukan (*Input*), keluaran (*Output*), pengolah (*Proses*), dan sasaran (*Objectives*), dan tujuan (*Goal*).”

#### 1. Komponen sistem (*components*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk satu kesatuan. Komponen-komponen sistem atau elemen-elemen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem. Setiap subsistem mempunyai sifat-sifat dari sistem untuk menjalankan suatu fungsi tertentu dan mempengaruhi suatu sistem secara keseluruhan.

#### 2. Batas Sistem (*boundary*)

Batas sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lain atau dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem di pandang sebagai satu kesatuan. Batas suatu sistem menunjukkan ruang lingkup dari sistem tersebut.

#### 3. Lingkungan Sistem (*environment*)

Lingkungan luar dari sistem adalah apapun diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan (harus dijaga dan merupakan energi dari sistem) dan dapat bersifat merugikan (harus ditahan dan dikendalikan).

#### 4. Penghubung Sistem (*interface*)

Penghubung merupakan media antara satu subsistem dengan subsistem yang lainnya. Melalui penghubung ini memungkinkan sumber- sumber daya mengalir dari subsistem ke subsistem yang lainnya. Keluaran (*output*) dari satu subsistem akan menjadi masukan (*input*) untuk subsistem yang lainnya dengan melalui penghubung. Dengan penghubung satu subsistem dapat berintegrasi dengan subsistem yang lainnya membentuk satu kesatuan.

#### 5. Masukan Sistem (*input*)

Masukan adalah energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan (*maintenance input*) dan masukan sinyal (*signal input*). *Maintenance input* adalah energi yang masukan supaya sistem tersebut dapat beroperasi. *Signal input* adalah energi yang diproses untuk didapatkan keluaran.

#### 6. Keluaran Sistem (*output*)

Keluaran adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Keluaran dapat merupakan masukan untuk subsistem yang lain atau kepada supra sistem.

#### 7. Pengolah Sistem (*Process*)

Suatu sistem dapat mempunyai bagian pengolah yang akan merubah masukan menjadi keluaran. Sistem akuntansi akan mengolah data-data transaksi menjadi laporan-laporan keuangan dan laporan-laporan lain yang dibutuhkan oleh manajemen.

## 8. Sasaran Sistem (*Objectives*)

Suatu sistem pasti mempunyai tujuan (*goal*) atau sasaran (*objective*). Kalau suatu sistem tidak mempunyai sasaran, maka operasi sistem tidak akan ada gunanya. Sasaran dari sistem sangat menentukan sekali masukan yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuan.

### 2.1.3. Klasifikasi Sistem

Adapun klasifikasi sistem menurut (Hutahaean, 2015:6) diuraikan sebagai berikut:

#### 1. Sistem Abstrak dan Sistem Fisik

Sistem abstrak merupakan 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 sehingga setiap mahluk dapat melihatnya, misalnya sistem komputer.

#### 2. Sistem Alamiah dan Sistem Buatan Manusia

Sistem alamiah merupakan sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia, misalnya sistem tata surya, sistem galaksi, sistem 11 reproduksi dan lain-lain. Sedangkan sistem buatan manusia merupakan sistem yang dirancang oleh manusia. Sistem buatan yang melibatkan interaksi manusia, misalnya sistem akuntansi, sistem informasi, dan lain-lain.

#### 3. Sistem Deterministik dan Sistem Probabilistik

Sistem deterministik merupakan sistem yang 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 robabilistik merupakan sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilitas, misalnya sistem manusia.

#### 4. Sistem Terbuka dan Sistem Tertutup

Sistem terbuka merupakan sistem yang berhubungan dan terpengaruh dengan lingkungan luarnya. Lebih spesifik dikenal juga yang disebut dengan sistem terotomasi, yang merupakan bagian dari sistem buatan manusia dan berinteraksi dengan kontrol oleh satu atau lebih komputer sebagai bagian dari sistem yang digunakan dalam masyarakat modern. Sistem ini menerima masukan dan menghasilkan keluaran untuk subsistem lainnya, misalnya sistem kebudayaan manusia. Sedangkan sistem tertutup merupakan sistem yang tidak berhubungan dan tidak terpengaruh dengan lingkungan luarnya. Sistem ini bekerja secara otomatis tanpa campur tangan dari pihak luar. Secara teoritis sistem tersebut ada, tetapi kenyataannya tidak ada sistem yang benar-benar tertutup, yang ada hanyalah *relatively closed system* (secara relatif tertutup, tidak benar-benar tertutup).

#### 2.1.4. Konsep Dasar Informasi

Pengertian menurut Krismaji (2015:14), Informasi adalah “data yang telah diorganisasi dan telah memiliki kegunaan dan manfaat”.

Hal serupa disampaikan oleh Romney dan Steinbart (2015:4) :

Informasi (*information*) adalah data yang telah dikelola dan diproses untuk memberikan arti dan memperbaiki proses pengambilan keputusan. Sebagaimana perannya, pengguna membuat keputusan yang lebih baik

sebagai kuantitas dan kualitas dari peningkatan informasi. Berdasarkan pengertian diatas dapat disimpulkan bahwa pengertian informasi adalah data yang diolah agar bermanfaat dalam pengambilan keputusan bagi penggunanya.

Dari uraian tentang informasi ini ada 3 hal penting yang harus di perhatikan disini, yaitu:

1. Informasi merupakan hasil pengolahan data
2. Memberikan makna atau arti
3. Berguna atau bermanfaat dalam meningkatkan kepastian

Menurut Pratama (2014:9) mengemukakan bahwa “Informasi merupakan hasil pengolahan data dari satu atau berbagai sumber, yang kemudian diolah, sehingga memberikan nilai, arti, dan manfaat.”

Berdasarkan defisini tersebut dapat disimpulkan bahwa informasi merupakan hasil pengolahan data dari berbagai sumber yang dapat memberikan manfaat, makna, atau arti bagi seseorang.

Selain memeberikan manfaat atau arti infomasi juga harus memiliki kualitas. Menurut Ardana dan Lukman (2016:11) patokan untuk menilai kualitas infomasi diantaranya:

1. Relevan (*Relevance*), informasi yang relevan akan mengurangi ketidak pastian, mengembangkan kemampuan pengambil keputusan untuk membuat prediksi atau mengkonfirmasi atau mengkoreksi ekspektasi sebelumnya.

2. Andal (*Reliable*), informasi yang handal jika informasi itu bebas dari kesalahan atau bias dan secara akurat menggambarkan kejadian yang terjadi dalam organisasi.
3. Lengkap (*Complete*), informasi yang lengkap jika tidak menghilangkan aspek-aspek penting dari peristiwa yang mendasari atau kegiatan yang terukur.
4. Tepat waktu (*Timely*), informasi yang tepat waktu bila informasi diberikan pada waktu pengambilan keputusan membuat keputusan.
5. Dapat dimengerti (*Understandable*), informasi yang dapat dimengerti jika disajikan dalam format yang bermanfaat dan memenuhi persyaratan bagi penggunaannya.
6. Dapat diverifikasi (*Verifiable*), informasi yang dapat diverifikasi jika dua orang yang kompeten bertindak secara independen akan menghasilkan informasi yang sama.
7. Dapat diakses (*Accessible*), informasi dapat diakses jika informasi itu tersedia bagi pengguna ketika dibutuhkan dan dalam format yang sesuai.

#### **2.1.5. Konsep Dasar Sistem Informasi**

Menurut Sutabri dalam (Samsinar dan Putrianti, 2015) mengemukakan bahwa : Sistem Informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk menyediakan laporan-laporan yang diperlukan oleh pihak luar tertentu.

Menurut Laudon dalam Ardana dan Lukman (2016:5) mendefinisikan: Suatu Sistem Informasi (SI) dapat didefinisikan secara teknis sebagai suatu rangkaian yang komponen-komponen saling terkait yang mengumpulkan (dan mengambil kembali), memproses, menyimpan dan mendistribusikan informasi untuk mendukung pengambilan keputusan dan mengendalikan perusahaan.

#### **2.1.6. Konsep Dasar Akuntansi**

Menurut Samryn (2014:3) menyimpulkan bahwa, “Secara umum akuntansi merupakan suatu sistem yang digunakan untuk mengubah data dari transaksi menjadi informasi keuangan”.

Menurut Bahri (2016:2) menyimpulkan bahwa, “Akuntansi adalah seni pencatatan, penggolongan, pengikhtisaran, dan pelaporan atas suatu transaksi dengan cara sedemikian rupa, sistematis dari segi isi, dan berdasarkan standar yang diakui umum”.

Dari teori di atas maka dapat diambil kesimpulan bahwa akuntansi adalah proses identifikasi, pencatatan, dan komunikasi terhadap transaksi-transaksi ekonomi menjadi informasi keuangan.

#### **2.1.7. Konsep Dasar Sistem Informasi Akuntansi**

Menurut Ardana dan Lukman (2016:6) “Sistem Informasi Akuntansi merupakan sistem yang mengumpulkan, mencatat, menyimpan, dan memproses data sehingga menghasilkan informasi bagi para mengambil keputusan.”

Menurut Susanto (2017:80) “Sistem informasi akuntansi dapat didefinisikan sebagai kumpulan (integrasi) dari sub-sub system atau komponen baik fisik maupun nonfisik yang saling berhubungan dan bekerja sama satu sama

lain secara harmonis untuk mengolah data transaksi yang berkaitan dengan masalah keuangan menjadi informasi keuangan.”.

### 2.1.8. Konsep Dasar Persediaan

Menurut Heizer dan Render (2015:553), “Persediaan adalah menentukan keseimbangan antara investasi persediaan dan pelayanan pelanggan. Tujuan persediaan tidak akan pernah mencapai strategi berbiaya rendah tanpa manajemen persediaan yang baik”

Menurut Assauri (2016:225) “Persediaan (*Inventory*) adalah stok dari suatu item atau sumber daya yang digunakan dalam suatu organisasi perusahaan.”

### 2.1.9. Metode Penilaian Persediaan

Metode Penilaian Persediaan Menurut (Meisak,2017) ada beberapa metode penilaian dalam persediaan yaitu:

a. FIFO (*First In First Out*)

Metode penetapan harga pokok persediaan yang di dasarkan atas anggapan bahwa barang-barang terlebih dulu di beli akan merupakan barang yang di jual pertama kali. Dalam metode ini persediaan akhir di nilai dengan harga pokok pembelian yang paling akhir.

b. LIFO (*Last In First Out*)

Metode ini kebalikan dari metode FIFO yang mengansumsikan bahwa penetapan harga pokok persediaan yang di dasarkan atas anggapan bahwa barang-barang yang paling akhir di beli akan merupakan yang pertama kali di jual. Dalam metode ini persediaan akhir akan di nilai dengan harga pembelian yang terdahulu.

c. Rata-rata (*Average*) Metode penetapan harga pokok persediaan di mana di anggap bahwa harga pokok rata-rata dari barang yang tersedia di jual akan di gunakan untuk menilai harga pokok yang di jual dan yang terdapat pada persediaan.

d. Indetifikasi Khusus (*Specific Identification*)

Dalam metode ini, harag pokok yang di bebaskan ke barang-barang yang di jual dan yang masih ada dalam persediaan di dasarkan atas harga poko yang di keluarkan khusus untuk barang-barang yang bersangkutan. Metode ini dalam praktek, hanya cocok untuk barang-barang yang jumlahnya tidak banyak dan nilai dari satuan barang nya tinggi, seperti misal nya mobil bekas dan lukisan.

e. Eceran

Metode penetapan harga pokok persediaan secara taksiran yang di dasarkan atas dasar hubungan, yang terdapat dalam tahun berjalan, antara harga pokok dengan harga jual.

f. Laba Bruto dan Laba Kotor (*gross profit method*)

Metode penetapan harga pokok persediaan secara taksiran yang di dasarkan atas hubungan, yang terdapat dalam periode yang lalu, antara laba bruto dengan harga jual. Metode ini hampir sama dengan metode eceran namun perbedaan terletak dalam penentuan persentase.

#### **2.1.10. Konsep Dasar Perangkat Lunak**

Menurut Sukamto dan Shalahudin (2016:22) mendefinisikan “Perangkat lunak adalah program komputer yang tersosiasi dengan dokumentasi perangkat

lunak seperti dokumentasi kebutuhan, model desain, dan cara penggunaan (*user manual*).”

### 2.1.11. Sistem Operasi

Menurut Kadir (2014:188) mengemukakan bahwa “Sistem operasi adalah program yang bertindak sebagai pengantar antara pemakai komputer dengan perangkat keras komputer.”

Tujuan dari sistem operasi adalah menyediakan lingkungan yang memungkinkan pemakai dapat menjalankan program apapun dengan mudah.

### 2.1.12. Netbeans

Netbeans adalah aplikasi *Integrated Development Environment* (IDE) yang berbasis Java.

Menurut Nofriadi (2015:4) “Netbeans merupakan sebuah aplikasi *Integrated Development Environment* (IDE) yang Menggunakan Bahasa Pemrograman Java dari Sun Microsystems yang berjalan diatas swing”, sedangkan menurut [www.netbeans.org/index\\_id.html](http://www.netbeans.org/index_id.html) (2017:1) “NetBeans IDE adalah sebuah lingkungan pengembangan sebuah kaskas untuk pemrogram menulis, mengompilasi, mencari kesalahan dan menyebarkan program. Netbeans IDE ditulis dalam Bahasa Pemrograman Java, namun dapat mendukung bahasa pemrograman lain”.

Dari kedua pendapat diatas dapat ditarik kesimpulan bahwa Netbeans adalah suatu aplikasi untuk membuat suatu program yang dimana bahasa pemrograman yang di pakai adalah java, maupun bahasa pemrograman yang lain dengan mudahnya penggunaan aplikasi netbeans ini maka programmer sering

menggunakan aplikasi ini 10 sebagai wadah untuk membuat berbagai macam program yang dimana netbeans IDE ini bersifat *Open Source*.

### 2.1.13. MySQL

Menurut Buana (2014:2), “MySQL Merupakan database server yang paling sering digunakan dalam pemrograman PHP. MySQL digunakan untuk menyimpan data dalam database dan memanipulasi data-data yang diperlukan. Manipulasi data tersebut berupa menambah, mengubah, dan menghapus data yang berada dalam database.

Menurut Hidayatullah dan Jauhari (2015:180) “MySQL adalah salah satu aplikasi DBMS yang sudah banyak oleh para pemogram aplikasi web. Contoh DBMS lainnya adalah : PostgreSQL (*freeware*), SQL Server, MS Access dari Microsoft, DB2 dari IBM, Oracle dan Oracle Corp, Dbase, FoxPro, dan sebagainya”.

### 2.1.14. phpMyAdmin

Menurut Buana (2014:2), “phpMyAdmin adalah salah satu aplikasi yang digunakan untuk memudahkan dalam melakukan pengelolaan *database* MySQL. phpMyAdmin merupakan aplikasi web yang bersifat *opensource*.”

Menurut Nugroho (2013:1) “PHP MyAdmin adalah aplikasi manajemen *database server* MySQL berbasis web. Dengan aplikasi phpMyAdmin kita bisa mengelola database sebagai *root* atau juga sebagai *user* biasa, kita bisa membuat *database* baru, megelola *database* dan melakukan operasi perintah-perintah database secara lengkap seperti saat kita di MySQL Prompt”.

## 2.2. Peralatan Pendukung (*Tools System*)

Merupakan alat yang digunakan untuk menggambarkan bentuk logika model dari suatu sistem dengan menggunakan simbol-simbol, lambang-lambang, diagram-diagram yang menunjukkan secara tepat arti dan fungsinya. Adapun peralatan pendukung (*tools system*) yang dijelaskan sebagai model sistem yang akan dirancang adalah sebagai berikut :

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

Menurut Sukamto dan Shalahudin (2016:87) mendefinisikan bahwa: *Unified Modeling Language (UML)* merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. UML hanya berfungsi untuk melakukan pemodelan. Jadi penggunaan UML tidak terbatas pada metodologi tertentu pada kenyataannya UML paling banyak digunakan pada metodologi berorientasi objek.

Secara fisik, menurut Sukamto dan Shalahudin (2016:140) “UML adalah Sekumpulan spesifikasi yang di keluarkan oleh OMG .”UML terbaru adalah UML 2.3 yang terdiri dari 4 macam spesifikasi, yaitu *Diagram Interchange Specification*, *UML Infrastructure*, *UML Superstructure*, dan *Object Constraint Language (OCL)*.”

Berikut ini penjelasan singkat dari pembagian kategori UML sebagai berikut:

#### a. *Structure Diagram*

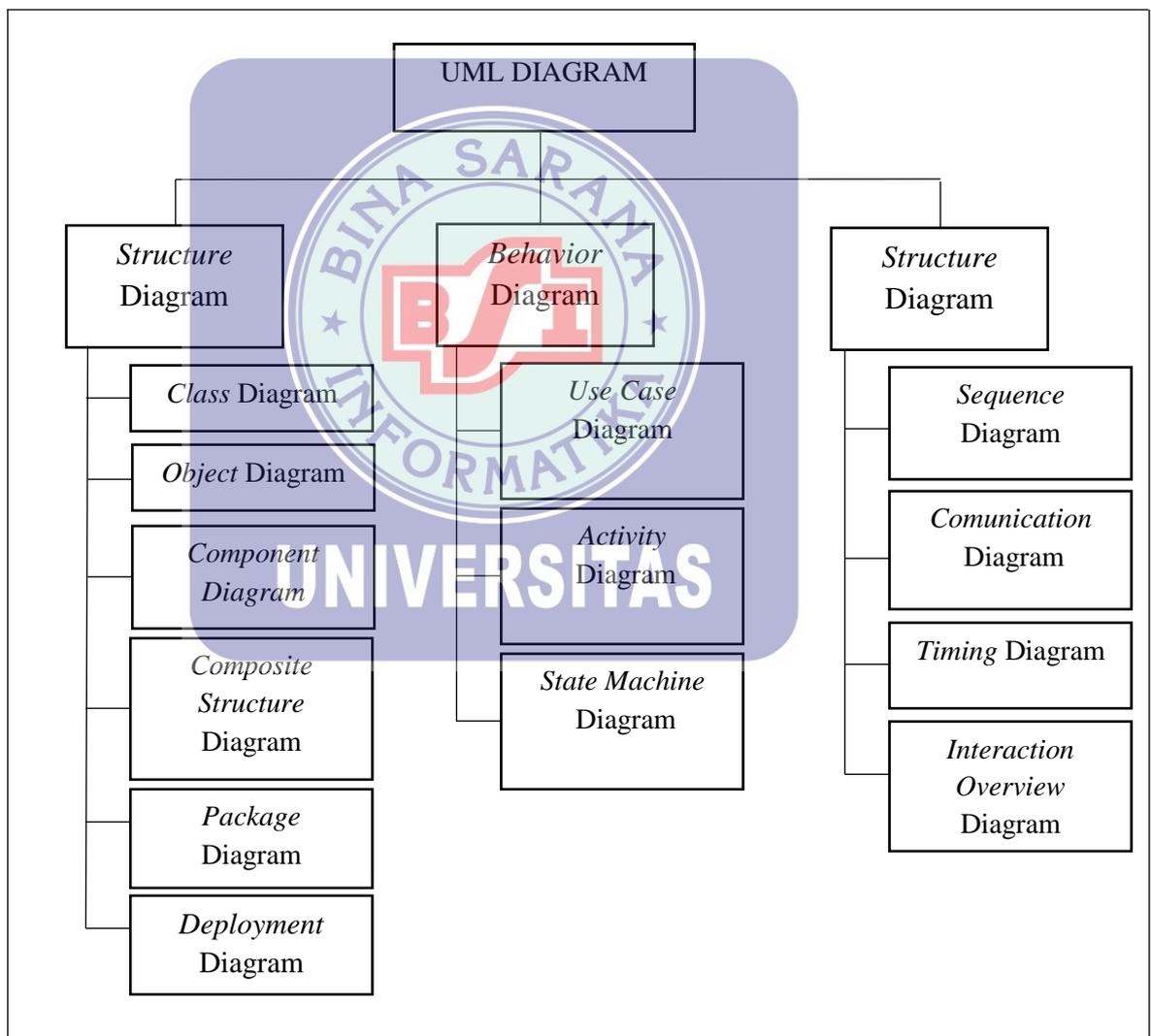
Yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan.

b. *Behavior Diagram*

Yaitu kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem.

c. *Interaction Diagram*

Yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antar subsistem pada suatu sistem.



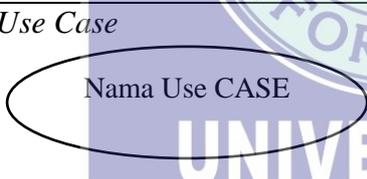
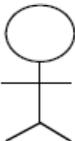
(Sumber: Sukamto dan Shalahuddin (2016:140))

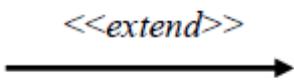
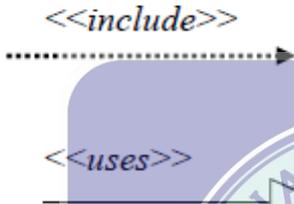
**Gambar 2.1**  
**Klasifikasi Diagram UML (Unified Modelling Language)**

Pada UML 2.3 terdiri dari 13 macam diagram, namun pada kesempatan kali ini penulis hanya akan membahas 4 macam diagram diantaranya, yaitu :

### 1. *Use Case Diagram*

Menurut Sukamto dan Shalahudin (2016:155) mengemukakan bahwa : *Use case* atau *diagram use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan di buat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. Adapun simbol-simbol yang digunakan dalam *use case* diagram adalah sebagai berikut :

No	Simbol	Deskripsi
1.	<p><i>Use Case</i></p> 	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>use case</i> .
2.	<p>Aktor / Actor</p>  <p>nama aktor nama_interface</p>	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama actor.
3.	<p>Asosiasi / <i>association</i></p> 	Komunikasi antar aktor dan use case yang berpartisipasi pada use case atau use case yang memiliki interaksi dengan actor.
4.	<p>Ekstensi / <i>extend</i></p>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu; mirip dengan prinsip

		<i>inheritance</i> pada pemrograman berorientasi objek; ditambahkan, missal arah panah mengarah pada <i>use case</i> yang ditambahkan; biasanya <i>use case</i> yang menjadi <i>extend</i> -nya merupakan jenis yang sama dengan <i>use case</i> yang menjadi induknya.
5.	Generalisasi/ <i>generalization</i> 	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya
6.	Menggunakan / include /uses 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> di mana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> .

(Sumber: Sukamto dan Shalahuddin (2016:156-158))

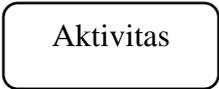
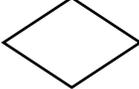
Gambar 2.2  
Simbol Activity Diagram

## 2. Activity Diagram

Sukamto dan Shalahuddin (2016:161) berpendapat bahwa “*Activity Diagram* dapat menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak.”

Adapun simbol-simbol yang digunakan dalam *activity diagram* adalah sebagai berikut :

No	Simbol	Deskripsi
1.	Status Awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.

2.	Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3.	Percabangan / <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
4.	Penggabungan / <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
5.	Status Akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
6.	<i>Swimlane</i> 	<i>Swimlane</i> memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

(Sumber: Sukamto dan Shalahuddin (2016:162-163))

**Gambar 2.3 Simbol  
Use Case Diagram**

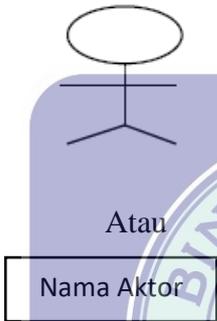
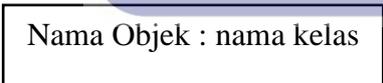
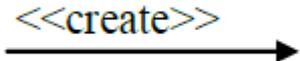
### 1. *Deployment Diagram*

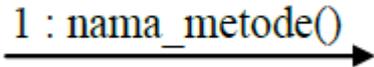
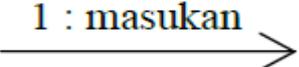
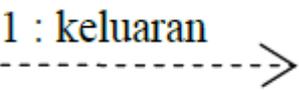
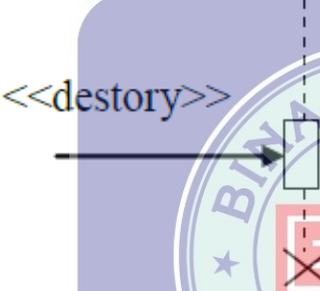
Menurut Sukamto dan Shalahudin (2016:154) berpendapat bahwa “Diagram deployment atau deployment diagram adalah menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi.”

### 2. *Sequence Diagram*

Menurut Sukamto dan Shalahudin (2016:165) mengemukakan bahwa : Diagram sekuen merupakan diagram yang dapat 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 use case beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Membuat diagram sekuen juga dibutuhkan untuk melihat skenario yang ada pada use case. Adapun simbol-simbol yang digunakan dalam *sequence* diagram adalah sebagai berikut :

No	Simbol	Deskripsi
1.	<p>Aktor</p>  <p>tanpa waktu aktif</p>	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama <i>actor</i> .
2.	<p>Garis hidup/ <i>lifeline</i></p> 	Menyatakan kehidupan suatu objek
3.	<p>Objek</p> 	Menyatakan objek yang berinteraksi pesan
4.	<p>Waktu aktif</p> 	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya
5.	<p>Pesan tipe <i>create</i></p> 	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat
6.	<p>Pesan tipe <i>call</i></p>	Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri.

		
7.	Pesan tipe <i>send</i> 	Menyatakan bahwa suatu objek mengirimkan data/ masukan/ informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.
8.	Pesan tipe <i>return</i> 	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian.
9.	Pesan tipe <i>destrory</i> 	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada <i>create</i> maka ada <i>destrory</i> .

(Sumber: Sukamto dan Shalahuddin (2016:165-167))

**Gambar 2.4**  
**Simbol Sequence Diagram**

### 2.2.2. Entity Relationship Diagram (ERD)

Menurut Pratama (2014:49) mengemukakan bahwa : ERD (*Entity Relationship Diagram*) adalah yang menggambarkan keterkaitan antartabel beserta field-field didalamnya pada suatu database sistem. Sebuah database memuat minimal sebuah tabel dengan sebuah atau beberapa buah field (kolom) didalamnya. Namun pada kenyataanya, database lebih sering dimiliki lebih dari satu buah tabel (dengan beberapa field didalamnya). Setiap tabel umumnya memiliki keterkaitan hubungan. Keterkaitan antartabel ini biasa disebut dengan relasi.

Terdapat tiga buah jenis relasi antar tabel didalam bagan ERD. Ketiga relasi tersebut yaitu:

1. *One to One* (Satu ke Satu)

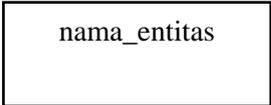
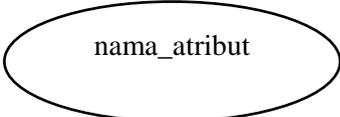
Relasi ini menggambarkan hubungan satu field pada tabel pertama ke satu field pada tabel kedua. Relasi ini paling sederhana. Sebagai contoh, pada sistem informasi perpustakaan terdapat tabel Buku (dengan field Kode\_Buku, Kode\_Kategori, Kode\_Penulis, Nama\_Penulis, Judul, Penerbit) dan tabel Kategori (Kode\_kategori , Nama\_Kategori, Alamat). Field Kode\_Kategori memiliki keterkaitan (relsi) satu ke satu pada tabel Buku dan tabel Kategori.

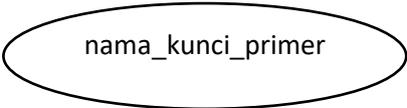
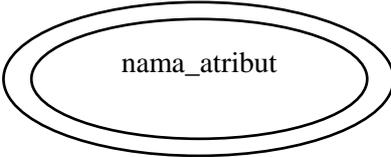
2. *One to Many* (Satu ke Banyak)

Relasi ini menggambarkan hubungan satu field pada tabel pertama kedua atau beberapa buah field ditabel kedua.

3. *Many to Many* (Banyak ke Banyak)

Adapun simbol-simbol yang digunakan dalam *Entity Relationship Diagram* adalah sebagai berikut :

SIMBOL	DESKRIPSI
Entitas / Entity 	Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data; benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi komputer; penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel
Atribut 	Field atau kolom data yang butuh disimpan dalam suatu entitas

<p>Atribut Kunci Primer</p> 	<p>Field atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses record yang diinginkan; biasanya berupa id; kunci primer dapat lebih dari satu kolom, asalkan kombinasi dari beberapa kolom tersebut dapat bersifat unik (berbeda tanpa ada yang sama)</p>
<p>Atribut multivalai/multivalue</p> 	<p>Field atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu</p>
<p>Relasi</p> 	<p>Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja</p>
<p>Asosiasi/association</p> 	<p>Penghubung antara relasi dan entitas dimana di kedua ujungnya memiliki multiplicity kemungkinan jumlah pemakaian Kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas yang lain disebut dengan kardinalitas. Misalkan ada kardinalitas 1 ke N atau sering disebut dengan one to many manghubungkan entitas A dan entitas B</p>

(Sumber : Sukamto dan Shallahudin (2016:50))

**Gambar 2.4 Simbol  
ERD (Entity Relationship Diagram)**

### 2.2.3. LRS (Logical Record Structured)

Menurut (Simarmata dan paryudi dalam Fridayanti dan Mahdiati 2016) “*Logical Record Structured* (LRS) adalah representasi dari struktur *record-record* pada tabel-tabel yang terbentuk dari hasil relasi antar himpunan entitas”.

Berikut adalah cara membentuk skema database atau LRS (*Logical Record Structured*) berdasarkan *Entity Relationship Diagram* :

1. Jika relasinya satu-ke-satu, maka *foreign key* diletakan pada salah satu dari dua entitas yang ada tau menyatukan kedua entitas tersebut.
2. Jika relasinya satu-ke-banyak, maka *foreign key* diletakan pada Entitas *Many*.
3. Jika relasinya banyak-ke-banyak, maka dibuat “file konektor” yang berisi dua *foreign key* yang berasal dari kedua entitas.

Menurut Frieyadie dalam (Isty dan Afifah, 2018) “LRS merupakan hasil dari pemodelan *Entity relationship* (ER) beserta atributnya sehingga bisa terlihat hubungan-hubungan antar entitas”.

#### 2.2.4. Spesifikasi File

Menurut (Wijaya dan Supriadi, 2015) mendefinisikana bahwa “Spesifikasi file adalah menjelaskan tentang file atau tabel yang terbentuk dari transformasi ERD (dan atau file-file penunjang untuk web)”. Berikut adalah contoh spesifikasifile admin, yaitu :

##### 1. Spesifikasi File Admin

Nama File : admin

Akronim : admin

Fungsi : untuk menyimpan data admin

Tipe File : File Master

Organisasi File : *Indexed Sequential*

Akses File : Random

Media : *Hardisk*

Panjang Record : 186

Kunci File : *number*

### Contoh Spesifikasi File Admin

No	Elemen Data	Tipe	Panjang	Ket
1.	<i>Number</i>	<i>Int</i>	1	PK
2.	<i>ID</i>	<i>Varchar</i>	20	
3.	<i>Password</i>	<i>Varchar</i>	50	
4.	<i>Email</i>	<i>Varchar</i>	50	
5.	<i>Full Name</i>	<i>Varchar</i>	50	
6.	<i>Phone</i>	<i>Varchar</i>	15	

Sumber : (Wijaya dan Supriadi, 2015) ★

**Tabel II.1.**  
**Spesifikasi File Admin**

#### 2.2.5. User Interface

Menurut Nurlifa dkk (2014:333) : “Setiap teknologi informasi memiliki interface atau antarmuka yang berfungsi untuk menjembatani antara pengguna dengan teknologi itu sendiri. Teknologi informasi yang satu dengan yang lain memiliki desain interface yang berbeda-beda sesuai dengan fungsi dan kebutuhan penggunanya”.

### 2.2.6. *Code Generation*

Menurut (Herliana & Rasyid, 2016) berpendapat bahwa “*Code generation* merupakan tahapan dilakukannya proses memasukan aturan dan kode dalam bentuk bahasa pemrograman.”

### 2.2.7. *Blackbox Testing*

Menurut Pratama (2014:50-51) berpendapat bahwa: *Blackbox Testing* merupakan pengujian di sisi pengembangan dilakukan oleh pengembangan/programmer dari aplikasi bersangkutan, atau mereka yang mengerti dan terlibat didalam pengembangan sistem tersebut. Pengujian ini menekankan pada sejauh mana fungsionalitas sistem informasi berjalan dengan baik sesuai dengan yang diinginkan oleh pengguna dan pengembang. Terdapat setidaknya empat buah jenis pengujian pada pengujian disisi pengembangan (*blackbox*) ini. Keempat pengujian tersebut meliputi:

1. Pengujian *Interface* (tatap muka) Aplikasi.

Pengujian interface (tatap muka) aplikasi sistem informais bertujuan untuk mengetahui fungsionalitas dari setiap elemen interface yang ada disetiap halaman pada aplikasi sistem informasi.

2. Pengujian Fungsi Dasar Sistem.

Pengujian fungsi dasar sistem bertujuan untuk mengetahui sejauh mana kinerja dari setiap fungsi dasar sistem informasi yang ada didalam aplikasi sistem infomasi.

### 3. Pengujian *Form Handle Sistem*

Pengujian *form handle* sistem bertujuan untuk mengetahui seperti apa dan sejauh mana respon oleh sistem informasi terhadap inputan yang diberikan oleh pengguna.

### 4. Pengujian Keamanan Sistem

Pengujian keamanan sistem bertujuan untuk mengetahui sejauh mana tingkat keamanan yang dimiliki oleh sistem informasi untuk dapat memberikan kenyamanan kepada para pengguna.

#### 2.2.8. Spesifikasi *Hardware* dan *Software*

Berikut adalah pengertian dari *hardware* dan *software*

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

Menurut Pratama (2014:2) berpendapat bahwa “Perangkat keras

(*Hardware*) merupakan perangkat yang mencakup semua perangkat keras komputer yang digunakan secara fisik didalam sistem informasi, baik di komputer server maupun di komputer client.”

##### 2. *Software* (Perangkat Lunak)

Menurut Pratama (2014:12) berpendapat bahwa “Perangkat lunak (*Software*) merupakan perangkat yang mencakup semua perangkat lunak yang digunakan didalam sistem informasi, yaitu: sistem operasi, aplikasi dan driver.”