

## **BAB II**

### **LANDASAN TEORI**

#### **2.1. Konsep Dasar Sistem**

Pada umumnya setiap organisasi mempunyai sistem informasi dalam mengumpulkan, menyimpan, melihat, dan menyalurkan informasi dalam membuat perancangan sistem informasi.

Konsep dasar sistem merupakan sekelompok komponen berbasis komputer yang dibuat oleh manusia dalam mengelola data, menyimpan, menghimpun kerangka kerja serta mengkoordinasikan sumber daya manusia dan komputer untuk mengubah sistem masukan menjadi sistem keluaran untuk mencapai tujuan dan sasaran yang telah ditetapkan sebelumnya.

##### **2.1.1. Pengertian Sistem**

Secara umum suatu sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur-unsur, komponen yang terorganisir, saling beinteraksi, saling ketergantungan satu sama lain dan terpadu untuk mencapai suatu tujuan tertentu.

Menurut (Hutahaean, 2015:2) mengemukakan bahwa “sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan kegiatan atau untuk melakukan sasaran yang tertentu”. Sedangkan menurut (Mulyani, 2016:2) menyatakan bahwa “sistem bisa diartikan sebagai sekumpulan sub sistem, komponen ataupun element yang saling bekerja sama dengan tujuan yang sama untuk menghasilkan output yang sudah

ditentukan sebelumnya”. Lain pula menurut (Marshall B. & Paul John, 2016 :3) “Serangkaian dua atau lebih komponen yang saling terkait dan berinteraksi untuk mencapai tujuan”.

Bersasarkan pengertian para ahli diatas maka dapat disimpulkan bahwa sistem adalah serangkaian elemen yang berkaitan dengan jaringan kerja yang saling berinteraksi satu dengan yang lainnya dan bertujuan untuk menangani dan menyelesaikan serangkaian kegiatan yang terjadi secara rutin dan berulang-ulang. Serangkaian elemen tersebut mempunyai arti bukan hanya elemen yang tampak secara fisik tetapi juga hal-hal yang bersifat abstrak atau konseptual seperti misi, pekerjaan, dan sebagainya.

### 2.1.2. Karakteristik Sistem

Suatu sistem mempunyai ciri-ciri karakteristik yang terdapat pada sekumpulan elemen yang harus dipahami dalam megidentifikasi pembuatan sistem. Adapun karakteristik sistem (Hutahaean, 2015:3) yang dimaksud adalah sebagai berikut:

#### 1. Komponen

Sistem terdiri dari sejumlah komponen yang saling berinteraksi dan bekerja sama untuk membentuk satu kesatuan. Komponen sistem dapat berupa sub sistem atau bagian-bagian dari sistem.

#### 2. Batasan sistem

Daerah yang membatasi antara suatu sistem dengan sistem lainnya atau dengan lingkungan luar dinamakan dengan batasan sistem. Batasan sistem ini

memungkinkan sistem dipandang sebagai satu kesatuan dan juga menunjukkan ruang lingkup (*scope*) dari sistem tersebut

### 3. Lingkungan luar sistem

Apapun yang berada di luar batas dari sistem dan mempengaruhi sistem tersebut dinamakan dengan lingkungan luar sistem. Lingkungan luar yang bersifat menguntungkan wajib dipelihara dan yang merugikan harus dikendalikan agar tidak mengganggu kelangsungan sistem.

### 4. Penghubung sistem

Media penghubung diperlukan untuk mengalirkan sumber-sumber daya dari sub sistem ke sub sistem lainnya dinamakan dengan penghubung sistem

### 5. Masukan sistem

Energi yang dimasukkan ke dalam sistem dinamakan dengan masukan sistem (*input*) dapat berupa perawatan dan masukan sinyal. Perawatan ini berfungsi agar sistem dapat beroperasi dan masukan sinyal adalah energi yang diproses untuk menghasilkan keluaran (*output*).

### 6. Keluaran sistem

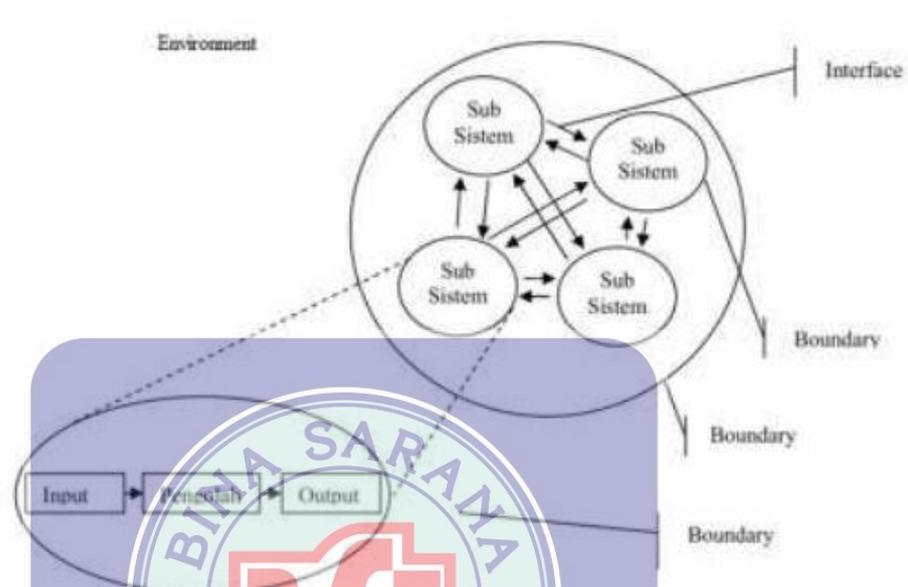
Hasil dari energi yang telah diolah dan diklasifikasikan menjadi keluaran yang berguna dinamakan dengan keluaran sistem (*output*). Informasi merupakan contoh keluaran sistem.

### 7. Pengolah sistem

Untuk mengolah masukan menjadi keluaran diperlukan suatu pengolah yang dinamakan dengan pengolah sistem

## 8. Sasaran sistem

Sistem pasti memiliki tujuan atau sasaran yang sangat menentukan input yang dibutuhkan oleh sistem dan keluaran yang dihasilkan



Sumber: Hutahaean (2015:4)

**Gambar II.1**  
**Karakteristik Sistem**

### 2.1.3. Klasifikasi Sistem

Sistem merupakan suatu bentuk integrasi antara satu komponen dan komponen lain karena sistem memiliki sasaran yang berbeda untuk setiap kasus yang terjadi di dalam sistem tersebut. Oleh karena itu sistem dapat diklasifikasikan dari beberapa sudut pandang. 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 makhluk 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 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 probabilistik 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. Pengertian Informasi**

Informasi mempengaruhi keberlangsungan dari organisasi bisnis atau pengguna informasi tersebut. Informasi akan menjadi berguna apabila objek yang menerima informasi tersebut membutuhkan informasi tersebut dan sesuai dengan harapan atau ekspektasi.

Menurut (Mulyani, 2016:17) mengemukakan bahwa “informasi merupakan data yang sudah diolah yang ditujukan untuk seseorang, organisasi ataupun siapa saja yang membutuhkan”. Sedangkan menurut (Marshall B. & Paul John, 2016:4) menyatakan bahwa “Data yang telah dikelola dan diproses untuk memberikan arti dan memperbaiki proses pengambilan keputusan.”. Begitu juga menurut (Hutahaean, 2015:9) mengemukakan bahwa “informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi penerimanya”.

Berdasarkan kutipan dari para ahli di atas, penulis menyimpulkan bahwa informasi merupakan kumpulan dari data yang telah diolah ke dalam bentuk yang lebih berguna bagi pengguna informasi yang menggambarkan suatu kejadian yang nyata dan dijadikan sebagai alat bantu untuk pengambilan keputusan.

### 2.1.5. Pengertian Sistem Informasi

Perusahaan atau lembaga sejenis yang mengikuti perkembangan zaman dan perkembangan teknologi menggunakan sistem informasi sebagai alat terapan di dalam organisasi atau perusahaan tersebut.

Menurut (Hutahaean, 2015:13) mengemukakan bahwa:

Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengelolaan transaksi harian, mendukung operasi, bersifat manajerial, dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang dibutuhkan.

Menurut (Eka, 2015:72) “Sistem informasi merupakan suatu sistem dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan informasi yang diperlukan untuk pengambilan keputusan”.

Berdasarkan dari pendapat di atas maka dapat diambil kesimpulan bahwa sistem informasi merupakan gabungan dari manusia, *hardware*, *software*, jaringan komunikasi dan data yang saling berinteraksi untuk menyimpan, mengumpulkan, memproses, dan mendistribusikan informasi untuk mendukung pengambilan keputusan dalam suatu organisasi.

### 2.1.6. Sistem Informasi Akuntansi

Menurut (Krismaji, 2018:4) “Sistem Informasi Akuntansi adalah sebuah sistem yang memproses data dan transaksi guna menghasilkan informasi yang bermanfaat untuk merencanakan, mengendalikan, dan mengoperasikan bisnis”.

Menurut (I.D.A.R.Tresnawati, 2017:1165) “Sistem informasi akuntansi adalah komponen organisasi yang dirancang untuk mengolah data keuangan menjadi informasi atau laporan keuangan, yang ditujukan kepada pihak internal

maupun eksternal perusahaan. Data dapat diolah menjadi informasi dengan cara manual maupun dengan bantuan komputer”.

Sementara menurut (Tmbooks, 2017:4) “Sistem Informasi Akuntansi (SIA) merupakan sistem yang mengumpulkan, mencatat, menyimpan dan memproses data sehingga menghasilkan informasi informasi bagi para pengambil keputusan.”

### **2.1.7. Pengertian Akuntansi**

Akuntansi adalah system informasi yang menghasilkan informasi keuangan kepada pihak –pihak yang berkepentingan mengenai aktivitas ekonomi dan kondisi suatu perusahaan.

Menurut (Hans Kartikahadi, 2016:3) pengertian akuntansi adalah : “Akuntansi adalah suatu sistem informasi keuangan, yang bertujuan untuk menghasilkan dan melaporkan informasi yang relevan bagi berbagai pihak yang berkepentingan”.

Menurut (Musta’an & H.K.Wardani, 2017:182) “Akuntansi adalah suatu sistem informasi yang mampu mengukur kegiatan atau aktivitas bisnis, melakukan kegiatan pemrosesan data sampai menjadi suatu laporan, serta mengkomunikasikan hasilnya kepada para pengambil keputusan dalam hal ini adalah para manajer”.

Dari definisi diatas maka dapat disimpulkan bahwa akuntansi adalah proses identifikasi, pencatatan, dan pengkomunikasian hasil akhir berupa laporan keuangan yang mencerminkan keadaan perusahaan kepada pihak-pihak yang berkepentingan.

### 2.1.8. Pengertian Pendapatan

M.Fuad,dkk. di dalam (Pastika, 2015:16) mendefinisikan bahwa “Pendapatan adalah perangkat suatu organisasi sebagai akibat dari penjualan barang dan jasa kepada pihak lain dalam periode akuntansi tertentu.

Menurut Rosjidi dalam (Handayani, 2016:9) menjelaskan bahwa “Pendapatan adalah peningkatan jumlah aktiva atau penurunan jumlah kewajiban perusahaan, yang timbul dari transaksi penyerahan barang dan jasa atau aktivitas usaha lainnya dalam suatu periode yang dapat diakui dan diukur berdasarkan prinsip akuntansi berlaku umum”.

Kesimpulan dari pendapat diatas menjelaskan bahwa “Pendapatan adalah penjualan barang dan jasa berdasarkan periode yang dapat diakui dan diukur berdasarkan prinsip akuntansi berlaku umum”.

### 2.1.9. Pengertian Jasa

Menurut Gronroos dalam (Nugraha, 2017:16) mendefinisikan jasa sebagai “Proses yang terdiri atas serangkaian aktivitas intangible yang biasanya (namun tidak harus selalu) terjadi pada interaksi antara pelanggan dan karyawan jasa dan atau sumber daya fisik atau barang dan atau sistem penyedia jasa, yang disediakan sebagai solusi atas masalah pelanggan”.

Jasa juga dikemukakan oleh Kotler dan Keller dalam buku (Tjiptono 2014:26) Jasa dapat didefinisikan sebagai “Setiap tindakan atau perbuatan yang dapat ditawarkan oleh suatu pihak pada pihak lain yang pada dasarnya bersifat intangible (tidak berwujud fisik) dan tidak menghasilkan kepemilikan sesuatu”.

(Noeraini & Sugiyono, 2016:7) Definisi diatas dapat disimpulkan bahwa “Jasa adalah sumber daya barang dan sistem penyedia jasa yang pada dasarnya tidak berwujud fisik dan tidak menghasilkan kepemilikan sesuatu”. Rumus Perhitungan Pendapatan Jasa/ Perusahaan Jasa, sebagai berikut :

- a. Persamaan Dasar Akuntansi :  $Harta = Utang + Modal$
- b. Laba Perusahaan Jasa :  $Laba = Pendapatan - Beban$
- c. Konsep Modal :  $Modal Akhir = Modal Awal + Laba - Prive$
- d. Perubahan Modal =  $Modal Akhir - Modal Awal$
- e. Siklus Akuntansi Jasa : Transaksi - Jurnal Umum - Buku Besar - Neraca Saldo - Jurnal Penyesuaian - Neraca Lajur - Laporan Keuangan - Jurnal Penutup - Neraca Saldo Penutup - Jurnal Pembalik.

Pencatatan pada jurnal jika terjadi transaksi atas pendapatan jasa adalah kas di debet dan pendapatan di kredit.

Pencatatan pada jurnal jika terjadi transaksi atas pendapatan jasa diterima sebagian uang atas pekerjaan yang telah selesai dikerjakan dan sebagian lagi diterima dikemudian, dalam jurnal dicatat sebagai berikut :

Kas	xxx	-
Piutang Usaha	xxx	-
Pendapatan	-	xxx

Pencatatan pada jurnal jika terjadi transaksi atas pendapatan jasa diterima pelunasan uang atas pekerjaan yang telah selesai dikerjakan, dalam jurnal dicatat sebagai berikut :

Kas	xxx	
Piutang Usaha	-	xxx

## 2.2. Peralatan Pendukung

Merupakan alat yang digunakan untuk menggambarkan logika dari suatu sistem dengan menggunakan simbol, lambang, diagram yang menunjukkan secara tepat arti dan fungsinya. Fungsi Tools system adalah untuk menjelaskan kepada user bagaimana fungsi dari sistem informasi dapat bekerja. Adapun peralatan pendukung yang dimaksud untuk merancang model sistem yang baru pada penulisan tugas akhir ini yaitu :

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

Menurut (Rosa & Shalahuddin, 2015:133) “UML muncul karena adanya kebutuhan pemodelan adanya kebutuhan visual untuk menspesifikasikan, menggambarkan, membangun dan dokumentasi dari system perangkat lunak”.

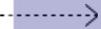
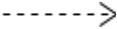
Sedangkan menurut (Rosa & Shalahuddin, 2015:134) UML (*Unified modeling Language*) adalah “salah satu standar Bahasa yang banyak digunakan didunia industri untuk mendefinisikan requirement, membuat analisis & desain, serta menggambarkan arsitektur dalam pemograman berorientasi objek”.

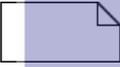
#### 2.2.1.1. *Use Case Diagram*

Menurut (Rosa & Shalahuddin, 2015:155) “ *Use Case* atau diagram *use case* merupakan pemodelan untuk kelakukan sistem informasi yang akan dibuat, *use case* mendeskripsikan sebuah interaksi antara satu atau lebih *actor* 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. Syarat penamaan pada *use case* adalah nama didefinikan sesimpel mungkin dan dapat dipahami. Dua hal utama pada *use case* yaitu pendefinsikan apa yang disebut *actor* dan *use case* :

1. Aktor Merupakan orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar system informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari *actor* adalah gambar orang, tapi *actor* belum tentu merupakan orang.
2. Use case merupakan fungsionalitas yang disediakan system sebagai unit-unit yang saling bertukar pesan antar unit atau actor.

**Tabel II.1**  
**Simbol Use Case Diagram**

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 eksplisit.
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 actor
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

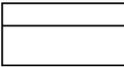
Sumber: Rosa dan Shalahuddin (2015:156)

### 2.2.1.2. Class Diagram

Menurut (Rosa & Shalahuddin, 2015:141)“Diagram kelas menggambarkan struktur system dari segi pendefinisikan kelas-kelas yang akan dibuat untuk membangun system:. Kelas memiliki apa yang disebut atribut dan metode atau operasi. Atribut merupakan variable-variabel yang dimiliki oleh suatu kelas, operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas.

Diagram kelas dibuat agar pembuat program atau programmer membuat kelas-kelas sesuai rancangan di dalam diagram kelas agar antara dokumentasi perancangan dan perangkat lunak sinkron.

**Tabel II.2**  
**Simbol Class Diagram**

NO	GAMBAR	NAMA	KETERANGAN
1		<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> ).
2		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor
5		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri ( <i>independent</i> ) akan memengaruhi elemen yang bergantung padanya elemen yang tidak mandiri
7		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

Sumber: Rosa dan Shalahuddin (2015:146)

### 2.2.1.3. Activity Diagram

Menurut (Rosa & Shalahuddin, 2015:161)“ Diagram aktivitas atau activity diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah *system* atau proses bisnis atau menu yang ada pada perangkat lunak”.

Perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas system bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh *system*.

**Tabel II.3**  
**Simbol Activity Diagram**

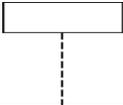
NO	GAMBAR	NAMA	KETERANGAN
1		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
4		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
5		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

Sumber: Rosa dan Shalahuddin (2015:162)

#### 2.2.1.4. Sequence Diagram

Menurut (Rosa & Shalahuddin, 2016:165), “Sequence diagram atau diagram sekuen menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan message yang dikirim dan diterima antar objek”. Oleh karena itu untuk menggambarkan *diagram sequence* maka harus diketahui objek-objek yang akan terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Membuat diagram sekuen juga dibutuhkan untuk melihat scenario yang ada pada *use case*.

**Tabel II.4**  
**Simbol *Sequence Diagram***

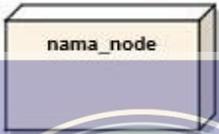
NO	GAMBAR	NAMA	KETERANGAN
1		<i>LifeLine</i>	Objek <i>entity</i> , antarmuka Yang saling berinteraksi.
2		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi
3		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi

Sumber: Rosa dan Shalahuddin (2015:166)

#### 2.2.1.5. Deployment Diagram

Menurut (Rosa & Shalahuddin, 2015:154) “*Diagram deployment* menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi”. *Diagram deployment* juga dapat digunakan untuk memodelkan hal-hal seperti, sistem tambahan (*embedded system*) yang menggambarkan rancangan *device*, *node*, dan *hardware*.

**Tabel II.5**  
**Simbol *Deployment Diagram***

Nama	Simbol	Deskripsi
<i>Package</i>		<i>Package</i> merupakan sebuah bungkusan dari satu atau lebih <i>node</i> .
<i>Node</i>		Biasanya mengacu pada perangkat keras ( <i>hardware</i> ), perangkat lunak yang tidak dibuat sendiri ( <i>software</i> ).
Kebergantungan ( <i>dependency</i> )		Kebergantungan antar <i>node</i> , arah panah mengarah pada <i>node</i> yang dipakai.
Link		Relasi antar <i>node</i> .

Sumber: Rosa dan Shalahuddin (2015:154)

### 2.2.2. *Enterprise Relationship Diagram (ERD)*

Pemodelan awal basis data yang paling banyak digunakan adalah menggunakan entity relationship diagram (ERD). ERD dikembangkan berdasarkan teori himpunan dalam bidang matematika.

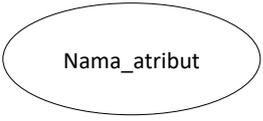
Menurut (Rosa & Shalahuddin, 2016:50) “entity relationship diagram (ERD) merupakan pemodelan awal basis data yang sering digunakan. ERD dikembangkan berdasarkan teori himpunan dalam bidang matematika”.

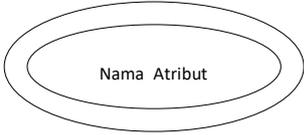
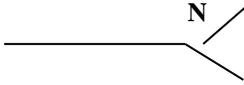
Sedangkan menurut (Lubis, 2016:31) “ERD menjadi salah satu pemodelan data konseptual yang paling sering digunakan dalam proses pengembangan basis data bertipe relasional”.

Dapat disimpulkan bahwa bahwa entity relationship diagram (ERD) merupakan pemodelan basis data konseptual dengan susunan data yang disimpan dalam sistem secara abstrak dengan menggunakan notasi dan simbol.

Simbol yang terdapat pada entity relationship diagram (ERD) sering disebut dengan komponen. Simbol pada entity relationship diagram (ERD) menurut Chen (Rosa & Shalahuddin, 2016:50) disajikan ke dalam bentuk tabel berikut ini.

**Tabel II.6**  
**Simbol Enterprise Relationship Diagram (ERD)**

No	Notasi	Komponen	Keterangan
1		Entitas	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.
2		Relasi	Relasi yang menghubungkan antar entitas, biasanya diawali dengan kata kerja
3		Atribut	Field atau kolom data yang butuh disimpan dalam suatu entitas.
4		Atribut kunci primer	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).
5		Atribut multivalai/multi value	Field atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki lebih dari satu.
6		Asosiasi/association	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 menghubungkan entitas A dan entitas B

Sumber: Rosa dan Shalahuddin (2015:51)

### 2.2.3. Logical Relational Structure (LRS)

*Logical record structure* (LRS) memiliki struktur record dari tabel yang ada di database yang direlasikan untuk mempermudah logika dari suatu program yang kita buat.

Menurut (Taufik, 2017:3) “sebelum tabel dibentuk dari field atau atribut entitas secara fisik atau level internal, maka harus dibuatkan suatu bentuk relational model yang dibuat secara logic atau level external dan konsep, dari pernyataan tersebut dibutuhkan yang disebut dengan *Logical Record Structure* (LRS)”

Berdasarkan pengertian menurut para ahli diatas dapat disimpulkan LRS (*Logical Record Structure*) dibutuhkan sebelum tabel dibentuk dari field atau atribut entitas secara fisik atau level external dan konsep. LRS sendiri merupakan representasi dari struktur record-record pada tabel-tabel yang terbentuk dari hasil kelas antar himpunan entitas pada diagram E-R.

#### 2.2.4. Basis Data

Basis data digunakan sebagai media penyimpanan data dan tempat pengolahan data menjadi informasi yang sangat penting dalam upaya menciptakan suatu aplikasi yang terintegrasi.

Menurut (Rosa & Shalahuddin, 2015:43) “sistem basis data adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah atau informasi dan membuat informasi tersedia saat dibutuhkan”.

Sedangkan, menurut (Lubis, 2016:2) mengemukakan bahwa “basis data merupakan gabungan file data yang dibentuk dengan hubungan/relasi yang logis dan dapat diungkapkan dengan catatan serta bersifat independen”.

Dapat ditarik kesimpulan bahwa basis data merupakan sekumpulan data yang tersimpan di tempat penyimpanan data dan dapat diolah menjadi informasi serta digunakan kembali jika dibutuhkan kembali Penulis menggunakan basis data dengan bahasa MySQL dan phpMyadmin sebagai aplikasi pengolah basis data.

##### 1. MySQL

Salah satu aplikasi basis data yang sering digunakan untuk mengolah dan menata file-file yaitu MySQL. Menurut (Manurung, 2015:39) “MySQL

merupakan turunan salah satu konsep utama basis data yang sudah ada sebelumnya SQL yaitu pngoperasian basisdata”.

Penulis menyimpulkan bahwa MySQL merupakan aplikasi pengolahan database yang sering digunakan untuk membuat sebuah aplikasi yang memiliki data-data sebagai sumber pengolahannya.

## 2. phpMyAdmin

Selain MySQL, aplikasi yang dapat mengolah basis data yaitu phpMyAdmin.

phpMyAdmin memiliki fungsi yang sama dengan MySQL, namun untuk pengaksesan aplikasi menggunakan browser.

Menurut (MADCOMS, 2016:186)“*PhpMyAdmin* adalah sebuah aplikasi Open Source yang berfungsi untuk memudahkan manajemen MySQL. Dengan menggunakan *PhpMyAdmin*, dapat membuat *database*, membuat tabel, meng-*insert*, menghapus dan meng-*update* data dengan GUI dan terasa lebih mudah, tanpa perlu mengetikkan perintah SQL secara manual”.

Berdasarkan pengertian diatas dapat disimpulkan bahwa PhpMyAdmin adalah sebuah aplikasi pemograman yang digunakan untuk manajemen database melalui browser (*web*) untuk mengontrol data mereka dan isi web yang akan ditampilkan dalam sebuah website yang mereka buat tanpa harus menggunakan perintah (*command*) SQL.

## 3. NetBeans

Menurut (Nofriadi 2015:4) dalam (Kaunen & Arizona, 2017) “Netbeans adalah sebuah aplikasi Integred Development Environment (IDE) yang berbasis java dari Sun Microsystems yang berjalan diatas swing dan banyak digunakan sebagai editor untuk berbagai pemograman”.

Definisi para ahli diatas NetBeans adalah aplikasi berbasis java yang berjalan diatas swing digunakan sebagai editor berbagai pemrograman dan dapat digunakan dengan gratis karena untuk keperluan yang didukung oleh Sun Microsystem”.

#### 4. XAMPP

Menurut Raharjo (2018:20) didalam (Syahrul.S.R & Desmulyati, 2019:24) menyatakan bahwa“ XAMPP adalah *software* yang membungkus *Apache HTTP Server*, *MariaDB*, *PHP* dan *Perl*. Dengan menggunakan XAMPP installasi paket *software* yang dibutuhkan untuk proses pengembangan *web* (*Apache HTTP Server*, *MariaDB* dn *PHP*) dapat dilakukan dengan sangat mudah, tanpa harus dilakukan secara terpisah (sendiri-sendiri)”.

#### 2.2.5. Pengujian *Blackbox Testing*

Menurut (Rosa dan Shalahuddin, 2015:272) “Pengujian perangkat lunak adalah sebuah elemen topik yang memiliki cakupan luas dan sering dikaitkan dengan verifikasi (*verification*) dan validasi (*validasi*) (V&V)”.

Menurut Rosa dan Shalahuddin (2015:272) pola pengujian pada perangkat lunak adalah sebagai berikut :

1. Pengujian dimulai dari level komponen hingga integrasi antara komponen menjadi sebuah sistem.
2. Teknik pengujian berbeda-beda sesuai dengan berbagai sisi atau unit uji dalam waktu yang berbeda-beda pula tergantung pada pengujian pada bagian mana yang dibutuhkan.

3. Pengujian dilakukan oleh pengembang perangkat lunak, dan jika untuk proyek besar, pengujian bisa dilakukan oleh tim uji yang tidak terkait dengan tim pengembang perangkat lunak (*independent test group (ITG)*).
4. Pengujian dan penirkutuan (*debugging*) merupakan aktifitas yang berbeda, tapi penirkutuan (*debugging*) harus diakomodasi pada berbagai strategi pengujian. Pengujian lebih fokus untuk mencari adanya kesalahan (*error*) baik dari sudut pandang pengembang tanpa harus menemukan lokasi kesalahan pada kode program. penirkutuan (*debugging*) adalah proses mencari lokasi kesalahan (*error*) pada kode program sehingga dapat segera diperbaiki oleh pembuat program (*programmer*).

Menurut (Sukamto dan Shalahuddin, 2016:274) pengujian untuk validasi memiliki beberapa pendekatan adalah sebagai berikut :

1. *Black Box Testing* (Pengujian Kotak Hitam)

Yaitu menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program.

2. *White Box Testing* (Pengujian Kotak Putih)

Yaitu menguji perangkat lunak dari segi desain dan kode program apakah mampu menghasilkan fungsi-fungsi, masukan, dan keluaran yang sesuai dengan spesifikasi kebutuhan.

Dalam tugas akhir ini penulis hanya menggunakan Pengujian Kotak Hitam atau *Black Box Testing*.

### 2.2.6. Spesifikasi Dokumen dan *File*

Menurut (Haryadi & Rita, 2016:374) “*Spesifikasi file* menjelaskan tentang *file-file* yang akan digunakan dalam rancangan program yang diajukan dan sebagai media penyimpanan data dari proses yang terjadi pada rancangan program”.

Dari beberapa pendapat diatas dapat disimpulkan bahwa spesifikasi file adalah penjelasan tentang *file-file* yang dibutuhkan dan digunakan dalam perancangan suatu program.

### 2.2.7. *Code Generation*

Menurut (Heru & Windha, 2016:2)“ Generator kode (*Code Generator*) adalah sebuah program aplikasi yang menulis kode program lainnya, atau dengan kata lain aplikasi yang menghasilkan aplikasi lain. Dengan semakin kompleksnya framework yang digunakan, kemampuan dalam mengembangkan sebuah generator kode sangat penting untuk membangun aplikasi.

*Code Generator* adalah “sebuah program yang dibuat agar dapat menghasilkan instruksi atau kode program sesuai dengan kebutuhan pengguna dalam ruang lingkup tertentu”

Dari defisi-defisi diatas dapat disimpulkan bahwa *code generation* adalah sebuah progam yang dibuat agar dapat menghasilkan instruksi atau kode progam sesuai dengan kebutuhan pengguna dalam ruang lingkup tertentu.

### 2.2.8. *Spesifikasi Hardware dan Software*

Menurut (Rosa & Shalahuddin, 2015:2) “Perangkat Lunak (*Software*) adalah program computer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain, dan cara penggunaan (*user manual*)

Tugas akhir ini dibuat dengan menggunakan Laptop dengan spesifikasi hardware sebagai berikut:

- a. Processor Intel Core i.3
- b. Memori 500 Gb
- c. Ram 2 Gb
- d. Monitor
- e. *Mouse*

Sistem operasi dan software yang digunakan dalam pembuatan tugas akhir ini adalah sebagai berikut :

- a. Bahasa *Script programming* : *Java NetBeans IDE 8.1*
- b. *Web Server* : Apache 3.2
- c. DBMS : MySQL
- d. JDK (*Java Development Kit*) versi 1.8
- e. Xampp Control Panel
- f. *Browser (Mozilla Firefox, Google Chrome)*

### 2.2.9. *User Interface*

Menurut (Mauladi & Tri, 2016:64) “Pengertian *User Interface* adalah salah satu layanan yang disediakan sistem operasi sebagai sarana interaksi antara pengguna dengan sistem operasi”.

Menurut (Mauladi & Tri, 2016:67) “Merupakan suatu antarmuka yang memungkinkan seseorang bisa berinteraksi dengan suatu informasi digital lewat lingkungan fisik”.

