

## **BAB II**

### **LANDASAN TEORI**

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

Dalam mendefinisikan sistem terdapat dua kelompok pendekatan sistem, yaitu sistem yang lebih menekankan pada prosedur dan elemennya. Prosedur didefinisikan sebagai suatu urutan yang tepat dari tahapan-tahapan intruksi yang menerangkan apa yang harus dikerjakan, siapa yang mengerjakan, kapan dikerjakan, dan bagaimana mengerjakannya. Pendekatan sistem yang merupakan kumpulan dari komponen atau elemen-elemen atau subsistem-subsistem merupakan definisi yang lebih luas dibandingkan pendekatan sistem yang lebih menekankan pada prosedurnya. Pendekatan sistem yang lebih menekankan pada komponen akan lebih mudah dipelajari untuk analisis dan rancangan sistem (Ladjamudin, 2013:2).

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

Sebuah sistem dapat diartikan sebagai suatu alat yang mana berikut ini beberapa pendapat para ahli mengenai tentang sistem, menurut Sutabri, (2012:3) sistem dapat diartikan sebagai suatu kumpulan/himpunan dari unsur, komponen, atau variabel yang terorganisasi, saling berinteraksi, saling tergantung satu sama lain dan terpadu.

Menurut Pratama (2014:7) “Sistem didefinisikan sebagai sekumpulan prosedur yang saling berkaitan dan saling terhubung untuk melakukan suatu tugas bersama-sama”.

Menurut Fitzgerald, et,al dalam Puspitawati dan Sri Dewi Anggadini (2010:1) “Suatu Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu “.

Dari pendapat para ahli mengenai sistem, dapat disimpulkan bahwa, “Sistem merupakan bagian-bagian yang saling berkaitan yang terintegrasi dengan maksud yang sama untuk melakukan suatu kegiatan tertentu”.

### **2.1.2. Pengertian Sistem Informasi Akuntansi**

Sistem Informasi Akuntansi (SIA) didefinisikan sebagai suatu sistem yang berfungsi untuk mengorganisasi formulir, catatan dan laporan yang dikoordinasi untuk menghasilkan informasi keuangan yang dibutuhkan dalam pembuatan keputusan manajemen dan pimpinan perusahaan dan dapat memudahkan pengelolaan perusahaan (Puspitawati dan Sri Dewi Anggadini, 2010:57).

Menurut Kusriani dan Andri Koniyo (2007:10) “Sistem Informasi Akuntansi merupakan sistem informasi yang mengubah data transaksi bisnis menjadi informasi keuangan yang berguna bagi pemakainya”.

Dari pendapat para ahli mengenai Sistem Informasi Akuntansi, dapat disimpulkan bahwa, “Sistem Informasi Akuntansi merupakan sistem yang menyediakan informasi akuntansi dan keuangan yang dibutuhkan dalam pembuatan keputusan manajemen dan dapat memudahkan data transaksi bisnis menjadi informasi yang berguna bagi penggunanya”.

### **2.1.3. Pengertian Pembelian**

Menurut Bahri (2016:288) Menjelaskan bahwa pembelian barang dagang adalah kegiatan yang berhubungan dengan pembelian barang dagangan.

Menurut A. Hall (2009:25) “Pembelian adalah tanggung jawab untuk memesan persediaan dari berbagai pemasok ketika tingkat persediaan jatuh ke titik pemesanan ulang”.

Dari pendapat para ahli mengenai pembelian dapat di simpulkan bahwa, Pembelian adalah kegiatan yang berhubungan dengan pemesanan barang dari berbagai pemasok.

#### **2.1.4. Pengertian Penjualan**

Menurut siswosoediro (2008:61) “Penjualan langsung (*direct selling*) adalah metode penjualan barang dan/atau jasa tertentu melalui jaringan pemasaran yang dikembangkan oleh mitra usaha yang bekerja atas dasar komisi atau bonus atas penjualan kepada konsumen diluar lokasi eceran tetap”.

Menurut Marwan dalam Rejeki dkk (2011:152) menjelaskan bahwa penjualan adalah suatu yang terpadu untuk mengembangkan rencana-rencana strategis yang diarahkan pada usaha pemuasan kebutuhan dan keinginan pembeli, guna mendapatkan penjualan yang menghasilkan laba.

Dari pendapat para ahli mengenai penjualan, dapat disimpulkan bahwa penjualan adalah jaringan pemasaran yang dikembangkan oleh mitra usaha yang diarahkan pada pemuasan kebutuhan dan keinginan pembeli.

#### **2.1.5. Karakteristik Sistem**

Menurut buku Analisis dan Desain Sistem Informasi karangan Ladjamudin (2013:3), suatu sistem mempunyai karakteristik atau sifat-sifat tertentu, yaitu mempunyai komponen-komponen, batas sistem, lingkungan luar sistem, penghubung, masukan, keluaran, pengolah dan sasaran atau tujuan.

### 1. Komponen Sistem

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

### 2. Batasan Sistem

Batasan sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batas system ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan dan menunjukkan ruang lingkup dari sistem tersebut.

### 3. Lingkungan Luar Sistem

Lingkungan luar dari suatu sistem adalah apapun diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan juga merugikan. Lingkungan luar yang menguntungkan merupakan energi dari sistem dan dengan demikian harus dijaga dan dipelihara. Sedangkan lingkungan luar yang merugikan harus ditahan dan dikendalikan, jika tidak maka akan mengganggu kelangsungan hidup dari sistem.

### 4. Penghubung Sistem

Penghubung merupakan media yang menghubungkan antara satu subsistem dengan yang lainnya. Melalui penghubung ini kemungkinan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lainnya. Keluaran dari

satu subsistem akan menjadi masukan untuk subsistem lainnya melalui penghubung.

#### 5. Masukan Sistem

Masukan sistem adalah energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan dan masukan sinyal *maintenance input* adalah energi yang dimasukkan supaya sistem tersebut dapat berjalan.

#### 6. Keluaran Sistem

Keluaran sistem adalah energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran dapat merupakan masukan untuk subsistem yang lain.

#### 7. Pengolahan Sistem

Suatu sistem dapat mempunyai suatu bagian pengolah atau sistem itu sendiri sebagai pengolahnya. Pengolah yang akan merubah masukan menjadi keluaran.

#### 8. Sasaran Sistem

Suatu sistem mempunyai tujuan atau sasaran, kalau sistem tidak mempunyai sasaran maka sistem tidak akan ada. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuannya. Sasaran sangat berpengaruh pada masukan dan keluaran yang dihasilkan.

### **2.1.6. Klasifikasi Sistem**

Menurut Ladjamudin (2013:6) Sistem merupakan suatu bentuk integrasi antara satu komponen dengan komponen lainnya. Karena sistem memiliki sasaran yang berbeda untuk setiap kasus yang terjadi yang ada didalam sistem tersebut.

Oleh karena itu sistem dapat diklasifikasikan kedalam beberapa sudut pandang yaitu (Ladjamudin, 2013:6) :

1. Sistem diklasifikasikan sebagai sistem abstrak dan sistem fisik. Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik. Sedangkan sistem fisik merupakan sistem yang ada secara fisik.
2. Sistem diklasifikasikan sebagai sistem alamiah dan sistem buatan manusia. Sistem alamiah adalah sistem yang terjadi Karena proses alam tidak dibuat oleh manusia (ditentukan dan tunduk kepada kehendak sang pencipta alam). Sedangkan sistem buatan manusia adalah sistem yang dirancang oleh manusia.
3. Sistem diklasifikasikan sebagai sistem tertentu (*deterministic system*) dan sistem tak tentu (*probabilistic system*). Sistem tertentu beroperasi dengan tingkah laku yang sudah dapat diprediksi. Sistem tak tentu adalah sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilitas.
4. Sistem diklasifikasikan sebagai sistem tertutup dan sistem terbuka. Sistem tertutup merupakan sistem yang tidak berhubungan dan tidak terpengaruh dengan lingkungan luarnya. Sistem terbuka adalah sistem yang berhubungan dan terpengaruh dengan lingkungan luarnya.

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

Peralatan pendukung mempunyai pengertian sebagai media yang dibutuhkan oleh setiap *programmer* untuk membantu mempermudah dalam

pembuatan pembacaan logika dan algoritma program, serta membantu untuk mengetahui alur program yang dibuat (Fridayanthie, 2015:145)

Dalam merancang sebuah sistem, diperlukan adanya suatu peralatan yang dapat mendukung terciptanya sebuah rancangan. Adapun peralatan pendukung yang dimaksud untuk merancang model sistem yang baru pada penulisan Tugas Akhir ini adalah:

### **2.2.1. Object Oriented Analysis and Design (OOAD)**

Menurut Fatta (2007:38) menyatakan bahwa “OOAD adalah metode pengembangan sistem yang lebih menekankan objek dibandingkan dengan data atau proses”.

Ada beberapa ciri khas dari pendekatan ini, yaitu *object*, *inheritance*, dan *object class* menurut Fatta (2007:38)

1. *Object* adalah struktur yang mengenkapsulasi atribut dan metode yang beroperasi berdasarkan atribut-atribut. *object* adalah abstraksi dari benda nyata dimana data dan proses diletakkan bersama untuk memodelkan struktur dan perilaku dari objek dunia nyata.
2. *Object class* adalah sekumpulan objek yang berbagi struktur yang sama dan perilaku yang sama.
3. *Inheritance* merupakan property yang muncul ketika tipe entitas atau *object class* disusun secara hierarki dan setiap tipe entitas atau *object class* menerima dan mewarisi atribut dan metode dari pendahulunya.

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

ERD digunakan untuk pemodelan basis data relasional. ERD biasanya memiliki hubungan binary (satu relasi menghubungkan dua buah entitas).

Beberapa metode perancangan ERD menoleransi hubungan relasi ternary (satu relasi menghubungkan tiga buah relasi) atau N-ary (satu relasi menghubungkan banyak entitas), tapi banyak metode perancangan ERD yang tidak mengizinkan hubungan ternary atau N-ary (Rosa dan Shalahuddin, 2013:50).

Menurut Abdullah (2015:23) “ERD merupakan suatu model untuk menjelaskan hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi”.

Menurut Pratama (2014:49) “ERD (*Entity Relationship Diagram*) adalah diagram yang menggambarkan keterkaitan antartabel beserta dengan *field-field* didalamnya pada suatu database sistem”.

Terdapat tiga buah jenis relasi antar table didalam bagan ERD. Ketiga relasi tersebut yaitu (Pratama, 2014:49):

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

Relasi ini menggambarkan hubungan satu field pada tabel pertama ke satu field pada tabel ke dua. relasi ini paling sederhana.

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

Relasi ini menggambarkan hubungan satu *field* pada tabel pertama ke dua atau beberapa buah *field* di tabel kedua.

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

Relasi dalam hal ini berupa relasi N ke N, yang artinya satuan atau lebih *field* pada tabel pertama dapat dihubungkan ke satu atau lebih ke *field* pada tabel kedua dan membentuk tabel ketiga.

### 2.2.3. LRS (*Logical Record Structure*)

Menurut Frieyadie dalam Rahmayu (2007:13) “*Logical Record Structure* (LRS) merupakan hasil pemodelan *Entity Relational Ship* (ER) beserta atributnya sehingga bisa terlihat hubungan-hubungan antar entitas”.

Dalam pembuatan LRS terdapat 3 hal yang dapat mempengaruhi yaitu :

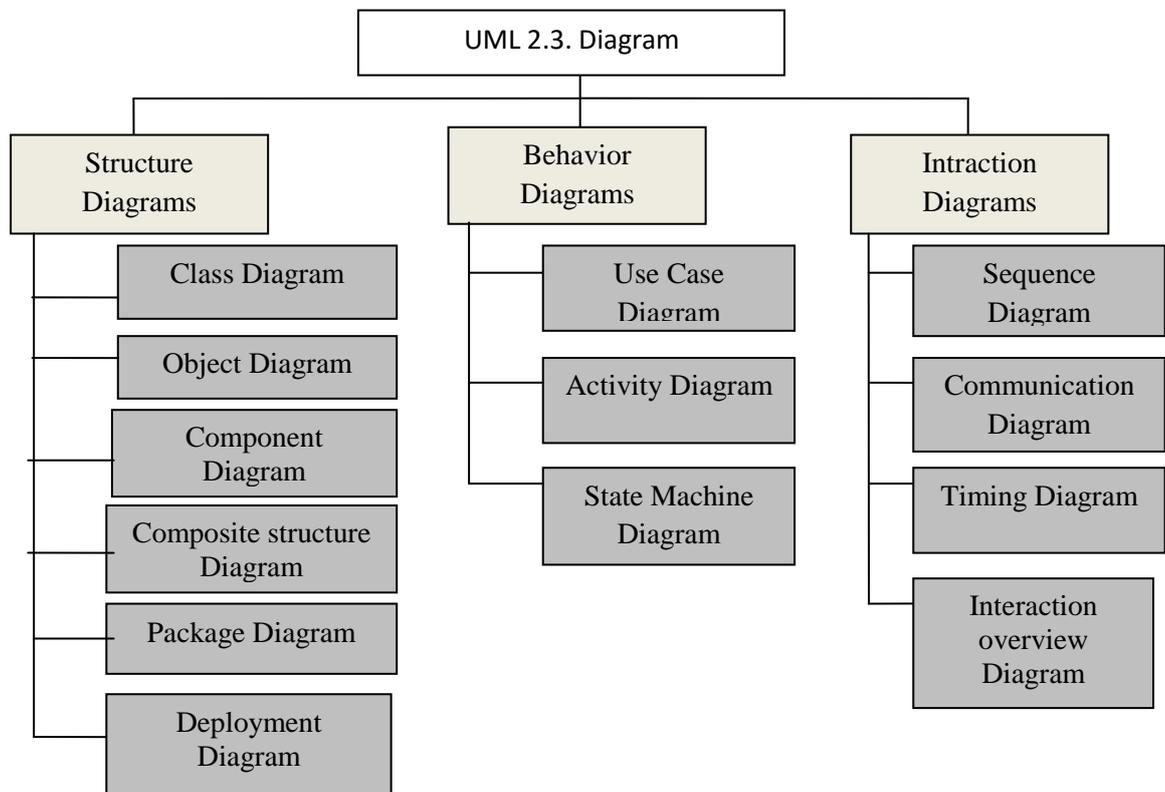
1. Jika tingkat hubungan (*cardinality*) satu pada satu (*one-to-one*), maka digabungkan dengan entitas yang lebih kuat (*strong entity*), atau digabungkan dengan entitas yang memiliki atribut yang lebih sedikit.
2. Jika tingkat hubungan (*cardinality*) satu pada banyak (*one-to-many*), maka hubungan relasi atau digabungkan dengan entitas yang tingkat hubungannya banyak.
3. Jika tingkat hubungan (*cardinality*) banyak pada banyak (*many-to-many*), maka hubungan relasi tidak akan digabungkan dengan entitas manapun, melainkan menjadi sebuah LRS.

### 2.2.4. Diagram UML (*Unified Modelling Language*)

Menurut Rosa A. S dan M. Shalahuddin (2013:140) “UML adalah sekumpulan spesifikasi yang dikeluarkan 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)”.

Pada UML 2.3 Terdiri dari 13 macam diagram yang dikelompokkan dalam 3 kategori.

Pembagian kategori dan macam-macam diagram tersebut dapat dilihat pada gambar dibawah ini (Rosa A. S dan M. Shalahuddin, 2013:140):



Sumber : Rosa dan Shalahuddin (2015:140)

**Gambar II.1 Diagram UML**

Berikut ini penjelasan singkat dari pembagian kategori tersebut.

1. *Structure Diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan.
2. *Behavior diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem.

3. *Interaction diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antar subsistem pada suatu sistem.

Menurut Pratama (2014:48) “UML (*Unified Modelling Language*) adalah standarisasi internasional untuk notasi dalam bentuk grafik, yang menjelaskan tentang analisis dan desain perangkat lunak yang dikembangkan dengan pemrograman berorientasi objek”.

Sebuah UML memiliki setidaknya Sembilan buah diagram di dalamnya, namun umumnya digunakan tiga buah diagram saja. Ketiga diagram tersebut mencakup *Use Case* diagram, *Class* Diagram, *Sequence* diagram.

1. *Use Case* diagram menggambarkan aliran kegiatan dan proses bisnis yang dilakukan oleh pengguna (aktor). pada diagram *Use Case* ini, setiap kegiatan digambarkan kedalam sebuah *Use Case* berbentuk lonjong dan terdapat minimal seorang aktor.
2. *Class* Diagram menggambarkan struktur kelas yang terdapat pada perangkat lunak yang akan dibangun. itu sebabnya UML digunakan untuk pemodelan aplikasi yang dibangun dengan konsep *object oriented* (Karena *Object oriented* kental dengan kelas, objek, pewarisan, method, dan sebagainya).
3. *Sequence* diagram menggambarkan sequence (aliran) pengiriman pesan (*message*) yang terjadi di aplikasi, sebagai bentuk interaksi dengan pengguna (*user*). *Sequence* diagram erat kaitannya dengan *Use Case* diagram.

Menurut Nugroho (2010:6) “UML (*Unified Modeling Language*) adalah bahasa pemodelan untuk sistem atau perangkat lunak yang berorientasi objek”.

## 1. *Use Case Diagram*

*Use Case* atau Diagram *Use Case* merupakan pemodelan untuk kelakuan (*Behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor sistem informasi yang akan dibuat. Secara kasar *use case* digunakan untuk mengetahui fungsi apa saja yang ada didalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. Syarat penamaan pada *use case* adalah nama didefinisikan sesimpel mungkin dan dapat dipahami. Ada dua hal utama pada *use case* yaitu pendefinisian apa yang disebut aktor dan *use case* (Rosa dan Shalahuddin, 2013:155).

- a. Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.
- b. *Use case* merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.

## 2. *Activity Diagram*

Diagram Aktivitas atau *activity* diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang biasa dilakukan oleh sistem (Rosa dan Shalahuddin, 2013:161).

Diagram aktivitas juga banyak digunakan untuk mendefinisikan hal-hal berikut (Rosa dan Shalahuddin, 2013:161):

- a. Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.
- b. Urutan atau pengelompokan tampilan dari sistem/user *interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan.
- c. Rancangan pengujian dimana setiap aktivitas dianggap memerlukan pengujian yang perlu didefinisikan kasus ujinya.
- d. Rancangan menu yang ditampilkan pada perangkat lunak.

### **3. Sequence Diagram**

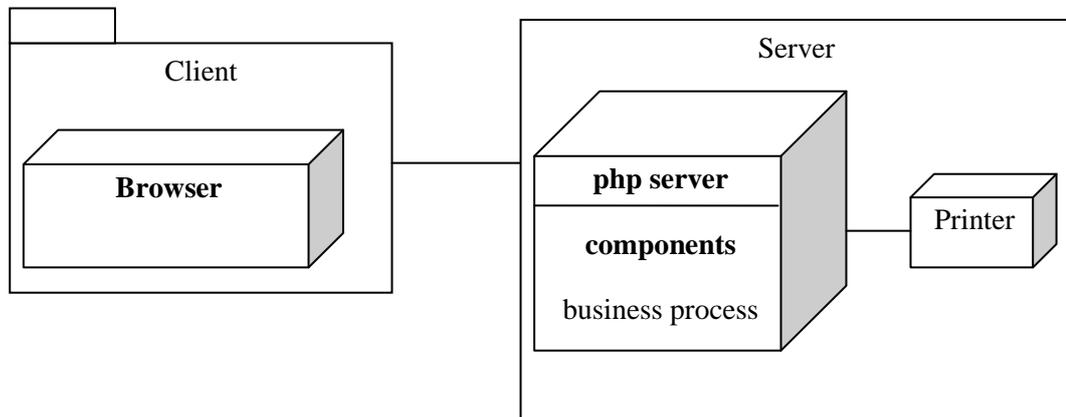
Menurut Rosa A. S dan M. Shalahuddin (2013:165) “Diagram sekuen menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan message yang dikirimkan dan diterima antar objek”.

Membuat diagram sekuen juga dibutuhkan untuk melihat skenario yang ada pada *use case*. Banyaknya diagram sekuen yang harus digambar adalah minimal sebanyak pendefinisian *use case* yang memiliki proses sendiri atau yang penting semua *use case* yang telah didefinisikan interaksi jalannya pesan sudah dicakup pada diagram sekuen sehingga semakin banyak *use case* yang didefinisikan maka diagram sekuen yang harus dibuat juga semakin banyak (Rosa A. S dan M. Shalahuddin, 2013:165).

### **4. Deployment Diagram**

Menurut Rosa A. S dan M. Shalahuddin (2013:154) Menjelaskan bahwa diagram *deployment* atau *deployment* diagram menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi. Diagram *deployment* juga dapat digunakan untuk memodelkan hal-hal berikut:

- a. Sistem tambahan (*embedded system*) yang menggambarkan rancangan device, node, dan hardware.
- b. Sistem *client/server* misalnya seperti gambar berikut:



Sumber : Rosa dan Shalahuddin (2015:154)

**Gambar II.2 Deployment Diagram**