

## **BAB II**

### **LANDASAN TEORI**

#### **II.1 Penelitian Terdahulu**

Penelitian ini dilakukan tidak terlepas dari hasil penelitian-penelitian terdahulu yang pernah dilakukan sebagai bahan perbandingan dan kajian. Adapun hasil-hasil penelitian yang dijadikan perbandingan tidak terlepas dari topik penelitian yaitu mengenai metode *Moving Average*. Penelitian sebelumnya dilakukan oleh Mulinda Oktaviani (2014) pada Direktorat *AEROSTRUCTURE* PT. Dirgantara Indonesia. Berdasarkan penelitian tersebut, metode pencatatan persediaan barang perusahaan menggunakan metode penilaian perpetual, dan metode penilaian persediaan yang ditetapkan adalah metode rata-rata bergerak (*moving average*). Sedangkan untuk pergerakan barang persediaan, perusahaan menggunakan metode *FIFO* (*First in First Out*). Penelitian lainnya dilakukan oleh Shinta Siti Sundari, dkk (2015) pada Toko The KIDS 24. Berdasarkan penelitian tersebut persediaan barang yang tidak tepat menimbulkan beberapa kerugian dari segi waktu dan juga biaya maka perlu adanya sebuah sistem peramalan. Peramalan merupakan suatu tehnik untuk mengidentifikasi suatu model yang dapat digunakan untuk meramalkan kondisi pada waktu yang akan datang. Dengan menggunakan metode *weight Moving Average* dapat diketahui nilai *error* lebih kecil dari metode lain dan hasil perkiraan dapat lebih tepat sehingga dapat membantu pemilik mengambil keputusan dalam melakukan persediaan barang.

## **II.2 Sistem Informasi**

Sistem terdiri dari bagian-bagian yang bersama-sama beroperasi untuk mencapai beberapa tujuan, dengan kata lain bahwa suatu sistem bukanlah merupakan suatu perangkat unsur-unsur yang dapat diidentifikasi sebagai kebersamaan yang menyatu disebabkan tujuan yang sama. (Sunyoto, 2014 : 33).

Informasi adalah data yang telah diolah menjadi sebuah bentuk yang berarti bagi penerimanya dan bermanfaat dalam mengambil keputusan saat ini atau mendatang. (Sunyoto, 2014 : 40)

Sistem informasi adalah sekumpulan komponen yang saling berhubungan, mengumpulkan atau mendapatkan, memproses, menyimpan dan mendistribusikan informasi untuk menunjang pengambilan keputusan dan pengawasan dalam suatu organisasi. (Sunyoto, 2014 : 47).

## **II.3 Akuntansi**

Akuntansi adalah sistem informasi yang menghasilkan laporan kepada pihak-pihak yang berkepentingan mengenai aktivitas ekonomi dan kondisi perusahaan. Pihak-pihak yang berkepentingan itu meliputi kreditor, pemasok, investor, karyawan, pemilik dan lain-lain. (Sunyoto, 2014 : 117)

Secara garis besar siklus akuntansi meliputi tahapan-tahapan sebagai berikut :

1. Mendokumentasi transaksi keuangan kedalam bukti Transaksi.
2. Mencatat Transaksi atas dasar bukti Transaksi ke dalam buku Jurnal.

3. Mengutip informasi dari buku Jurnal dan selanjutnya dipindahkan kedalam buku Besar.
4. Menentukan saldo-saldo buku Besar di akhir periode dan menuangkannya dalam neraca saldo.
5. Menyesuaikan buku Besar berdasarkan informasi yang terkini (*uptodate*)
6. Menentukan saldo-saldo buku Besar setelah penyesuaian dan menuangkannya dalam Neraca Saldo Setelah Penyesuaian (NSSP)
7. Menyusun laporan keuangan berdasarkan pada Neraca Saldo Setelah Penyesuaian (NSSP)
8. Menutup buku Besar.
9. Menentukan saldo-saldo buku Besar dan menuangkannya dalam Neraca Saldo setelah penutupan.

#### **II.4 Sistem Informasi Akuntansi**

Sistem informasi akuntansi adalah struktur yang menyatu dalam suatu entitas, yang menggunakan sumber daya fisik dan komponen lain, untuk mengubah data transaksi keuangan/akuntansi menjadi informasi akuntansi dengan tujuan untuk memenuhi kebutuhan akan informasi dari para pengguna atau pemakainya. (Sunyoto, 2014 : 118).

#### **II.5 Persediaan**

Persediaan adalah nama yang diberikan untuk barang-barang baik yang dibuat atau dibeli kembali dalam bisnis normal. Dalam perusahaan manufaktur

terdiri dari persediaan bahan mentah, persediaan pekerjaan dalam proses dan persediaan dalam bentuk barang jadi. (Setyanto, 2014 : 6)

Pada umumnya persediaan dinyatakan dalam neraca sebesar harga pokok atau harga perolehan yang bersangkutan, yang meliputi persediaan tersebut pada keadaan dan tempat sebagaimana adanya. Baik perusahaan dagang maupun industri dalam laporan tahunnya selalu menunjukkan bahwa unsur terbesar dalam aktiva lancar pada neraca adalah persediaan. Persediaan meliputi barang yang dibeli dan disimpan untuk dijual kembali. Persediaan mencakupi barang-barang yang telah diproduksi, atau barang dalam penyelesaian yang sedang diproduksi perusahaan, dan termasuk pula bahan serta perlengkapan yang dipergunakan dalam proses produksi.

Pengertian persediaan yang dimaksud oleh Standart Akuntansi Indonesia bila dibandingkan dengan persediaan yang diuraikan oleh para ahli lainnya terlihat adanya persamaan pengertian, hanya menurut PSAK dipertegas mengenai pengakuannya terhadap persediaan tersebut. Namun demikian pada prinsipnya adalah sama yaitu merupakan salah satu unsur dari aktiva lancar perusahaan yang dipergunakan sebagai objek perusahaan, baik itu berupa persediaan bahan baku, bahan pembantu, barang dalam proses produksi ataupun barang jadi yang siap dijual selama kegiatan normal perusahaan.

### **II.5.1 Metode Pencatatan Persediaan**

Pada pokoknya ada dua metode pencatatan persediaan yaitu Metode Fisik (periodik) dan Metode Perpetual (buku).

Dalam metode periodik, adanya transaksi pembelian tidak didebet pada rekening persediaan tapi didebet pada rekening pembelian begitu juga dengan transaksi penjualan tidak dikredit pada rekening persediaan tapi pada rekening penjualan. Informasi mengenai persediaan yang ada pada suatu saat tertentu, tidak didapat dari rekening persediaan tapi melalui perhitungan fisik atas persediaan yang ada digudang. Perhitungan fisik biasa dilakukan pada saat perusahaan akan menyusun laporan keuangan. Dalam metode ini perhitungan fisik mempunyai peranan penting, karena tanpa perhitungan fisik laporan keuangan tidak dapat disusun. Dalam pembahasan ini kita akan menggunakan metode fisik atau periodik.

Dalam metode perpetual, baik jumlah penjualan maupun harga pokok penjualan dan dicatat pada setiap saat barang dijual. Dengan cara ini catatan akuntansi akan secara terus menerus mengungkapkan besarnya persediaan yang ada.

### **II.5.2 Metode Penilaian Persediaan**

Persediaan memegang peranan yang penting untuk menentukan hasil usaha selama periode tertentu. Hal ini karena jumlah yang terjual dalam satu periode tidak sama dengan jumlah barang yang dibeli/diproduksi dalam periode yang sama. Disamping itu harga pokok barang yang dibeli/diproduksi pun dalam satu periode yang berbeda. Oleh karena itu maka perlu dilakukan penilaian persediaan untuk mengetahui besarnya nilai persediaan akhir.

Berikut ini diuraikan beberapa metode penilaian persediaan yang dapat digunakan pada metode pencatatan fisik dan perpetual :

#### **A. Metode Penilaian Persediaan Menurut Fisik**

Penilaian persediaan berdasarkan metode fisik terdiri dari :

##### 1. Identifikasi Khusus

Menurut cara ini, setiap barang yang dibeli perusahaan dimasukkan ke gudang penyimpanan, harus diberi (di stempel) tanda pengenal. Dalam tanda pengenal tersebut harus dicantumkan harga pembelian barang yang bersangkutan. Oleh karena setiap barang mempunyai tanda pengenal dan harganya, maka pada akhir periode untuk mengetahui nilai persediaan akhir barang tersebut, cukup dengan melihat dan memperhitungkan jumlah sisa barang itu beserta harga perolehannya.

##### 2. Rata-rata Sederhana

Menurut rata-rata sederhana, sisa persediaan barang dinilai berdasarkan rata-rata harga pembelian yang pernah terjadi dan dialami oleh perusahaan.

##### 3. Rata-rata Tertimbang

Menurut cara rata-rata tertimbang, sisa persediaan barang dinilai berdasarkan harga rata-rata dari seluruh pembelian yang pernah terjadi, yaitu nilai rupiah seluruh pembelian dibagi dengan seluruh kilogram barang yang dibeli. Misalnya barang-barang yang ada pada

tanggal 28 Februari 2001 dihitung berjumlah 300 kg, maka persediaan akhir barang dihitung sebagai berikut :

Feb 1	Persediaan	200 kg @ Rp. 100,-	= Rp. 20.000,-
09	Pembelian	300 kg @ Rp. 110,-	= Rp. 33.000,-
15	Pembelian	400 kg @ Rp. 116,-	= Rp. 46.400,-
24	Pembelian	100 kg @ Rp. 126,-	= Rp. 12.600,-
		1000 kg	= Rp. 112.000,-
Harga pokok rata-rata tertimbang :			
$\frac{\text{Rp. 112.000,-}}{1000 \text{ kg}} = \text{Rp. 112,- per kg}$			
Persediaan akhir barang tanggal 28 Februari 2001 :			
300 kg @ Rp. 112,- = Rp. 33.600,-			
Harga pokok penjualan :			
$\text{Rp. 112.000} - \text{Rp. 33.600} = \text{Rp. 78.400,-}$			

#### 4. *First In First Out (FIFO)*

Menurut cara ini (*FIFO*), harga barang yang masuk (dibeli) lebih awal dianggap dikeluarkan (dijual) lebih awal pula. Dengan demikian sisa persediaan barang pada akhir periode adalah barang-barang dengan harga yang masuknya (dibelinya) paling akhir. Misalnya perhitungan fisik atas barang-barang dalam gudang tanggal 28 Februari 2001 menunjukkan jumlah 300 kg. jumlah persediaan akhir ini terdiri dari :

Persediaan 24 Feb	100 kg @ Rp. 126,-	= Rp. 12.600,-
Pembelian 15 Feb	200 kg @ Rp. 116,-	= Rp. 23.200,-
300 kg		= Rp. 35.800,-
Harga pokok penjualan :		
$\text{Rp. 112.000} - \text{Rp. 35.800} = \text{Rp. 76.200,-}$		

#### 5. *Last In First Out (LIFO)*

Menurut cara ini (*LIFO*), harga barang yang masuk (dibeli) awal dianggap dikeluarkan (dijual) lebih akhir. Dengan demikian sisa persediaan barang pada akhir periode adalah barang-barang dengan harga yang masuknya (dibelinya) paling awal. Misalnya pada tanggal

28 Februari 2001 diadakan perhitungan fisik terhadap barang-barang dalam gudang yang hasilnya menunjukkan jumlah persediaan sebanyak 300 kg. Harga pokok persediaan dapat dihitung :

Persediaan 1 Feb	200 kg @ Rp. 100,-	= Rp. 20.000,-
Pembelian 9 Feb	100 kg @ Rp. 110,-	= Rp. 11.000,-
300 kg		= Rp. 31.000,-

Harga pokok penjualan :

$$\text{Rp. 112.000} - \text{Rp. 31.000} = \text{Rp. 81.000,-}$$

#### 6. *Gross Profit*

Menurut cara *Gross Profit* (laba bruto), sisa persediaan barang pada akhir periode dihitung dengan mengurangkan nilai barang yang siap untuk dijual pada akhir periode yang bersangkutan dengan harga beli barang yang terjual selama periode tersebut. Sedangkan nilai barang yang siap untuk ditentukan dengan menjumlahkan persediaan awal periode dengan jumlah pembelian netto yang dilakukan selama periode tersebut. Sedangkan nilai harga pokok barang yang dijual dihitung berdasarkan nilai penjualan bersih dikurangi dengan laba bruto yang diperoleh selama periode tersebut.

### **B. Metode Penilaian Persediaan Menurut Perpetual**

Penilaian persediaan berdasarkan metode perpetual terdiri dari :

#### 1. *First In First Out (FIFO)*

Menurut metode FIFO, barang yang dibeli atau masuk lebih awal, dianggap dikeluarkan lebih awal pula. Ini berarti setiap terjadi

transaksi penjualan, harga pokok barang yang terjual, dinilai berdasarkan pada harga barang yang dibeli lebih awal.

2. *Last In First Out (LIFO)*

Berdasarkan metode *LIFO*, barang yang dibeli paling akhir, dianggap dikeluarkan lebih awal. Dengan demikian setiap terjadi suatu transaksi penjualan, maka harga pokok barang yang terjual dinilai berdasarkan pada harga barang yang dibeli lebih akhir.

3. *Moving Average (Rata-rata Bergerak)*

Metode ini menetapkan bahwa setiap terjadi perubahan jumlah persediaan barang baik karena pembeli maupun karena penjualan maka sisa persediaan barang yang ada segera dihitung kembali nilai rata-ratanya. Nilai rata-rata yang masih ada diperoleh dengan cara membagi jumlah nilai persediaan barang yang masih ada dengan jumlah satuan barang yang bersangkutan. Dengan demikian harga pokok barang yang dijual, dinilai berdasarkan pada rata-rata terbaru.

## II.6 JAVA

Java merupakan bahasa pemrograman tingkat tinggi yang memiliki karakteristik *simple, object oriented, distributes, interpreted* dan memiliki performa yang tinggi. (Komputer, 2015 : 2).

Bahasa pemrograman java merupakan *compiler* sekaligus *interpreter*, dimana sebagai *compiler* program yang telah dibuat akan diubah menjadi *Java*

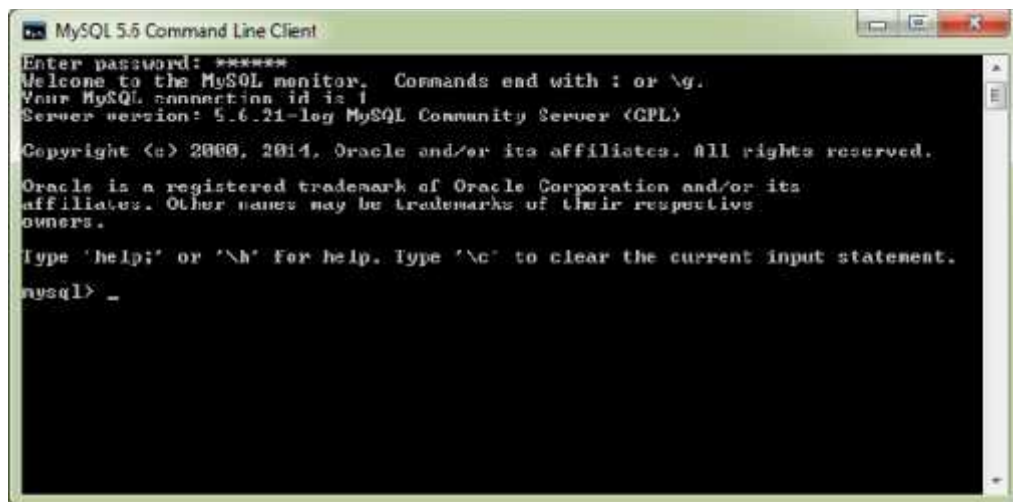
*Bytecodes*. Sedangkan sebagai *interpreter*, *Java Bytecodes* tersebut dijalankan pada komputer.

Pada bahasa pemrograman *Java* terdapat berbagai macam fitur yang disediakan oleh platform teknologi *Java*, yaitu *Java Virtual Machine (JVM)*, *Garbage Collection* dan *Code Security*.

## **II.7 MySQL**

*MySQL* adalah adalah salah satu jenis *database server* yang sangat terkenal di dunia. *MySQL* termasuk jenis *RDBMS (Relational Database Managemen System)*. Oleh karena itu, istilah seperti tabel, baris dan kolom banyak digunakan pada *MySQL*. (Komputer, 2015 : 6)

Contoh *RDBMS* lain adalah *Oracle*, *Sybase*. Basis data memungkinkan anda untuk menyimpan, menelusuri, menurutkan dan mengambil data secara efisien. *Server MySQL* yang akan membantu melakukan fungsionalitas tersebut. Bahasa yang digunakan oleh *MySQL* tentu saja adalah *SQL-standar* bahasa basis data relasional di seluruh dunia saat ini. Tampilan awal *MySQL* dapat dilihat pada gambar II.1 seperti berikut.



**Gambar II.1 Tampilan Awal MySQL**  
(Sumber : Komputer ; 2015)

## **II.8 UML (Unified Modelling Language)**

*UML* merupakan kependekan dari *Unified Modeling Language* yaitu diagram dan metode standar untuk memodelkan dan merepresentasikan *object oriented software* dan sistem bisnis. (Mulyani, 2016 : 243).

Beberapa fungsi dan kegunaan dari *UML* yaitu (Mulyani, 2016 : 244) :

### 1. *Visualizing*

*Visualizing* yaitu sebagai alat komunikasi konseptual model antara tim pengembang sistem (sistem analis dengan programmer)

### 2. *Specifying*

*Specifying* yaitu sebagai *tools* yang digunakan untuk memodelkan sistem secara tepat dan jelas.

### 3. *Constructing*

*Constructing* yaitu *UML* sebagai bahasa gratis mampu melakukan *mapping* dan konseptual model kedalam bahasa pemrograman.

#### 4. *Documenting*

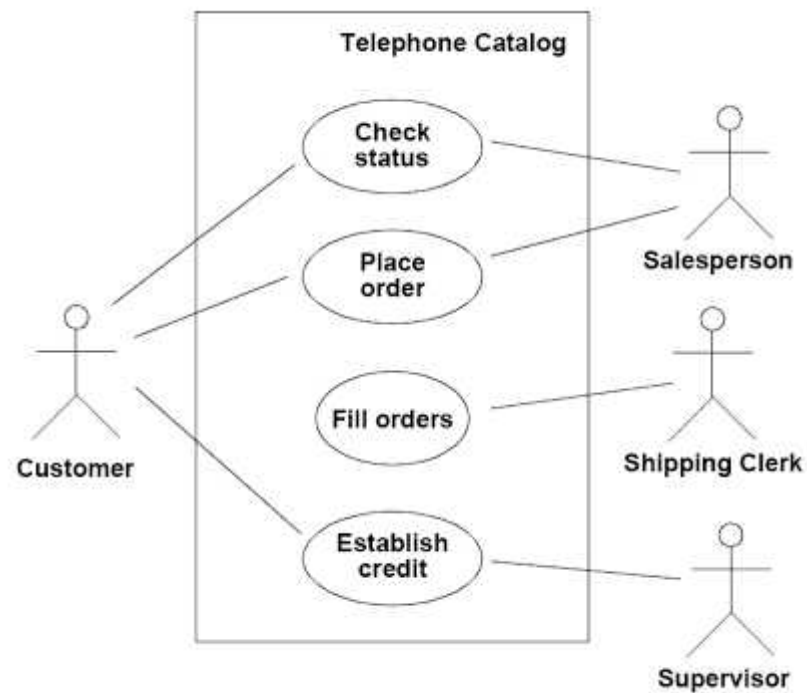
*Documenting* yaitu *UML* digunakan sebagai *tools* untuk melakukan dokumentasi teknis sebuah sistem.

Diagram-diagram yang terdapat dalam *UML* sangat banyak, berikut ini beberapa diagram yang sering digunakan dalam pengembangan sistem yaitu :

##### 1. Use Case Model

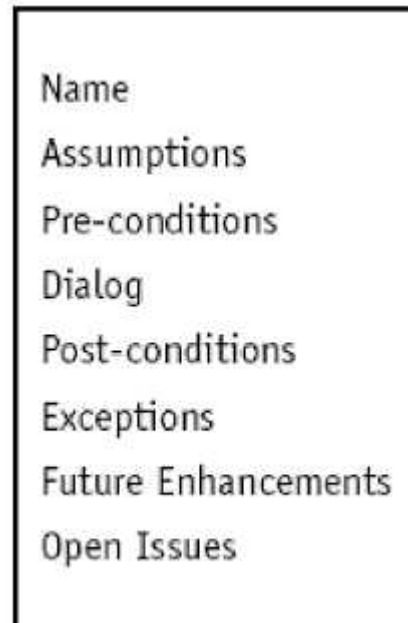
Use Case Model merupakan kumpulan diagram dan text yang saling bekerja sama untuk mendokumentasikan bagaimana user (aktor) berinteraksi dengan sistem. Use Case Model terdiri dari beberapa diagram:

1) *Use case diagram* yaitu diagram yang menggambarkan dan merepresentasikan aktor, *use cases*, dan *dependencies* suatu proyek dimana tujuan dan diagram ini adalah untuk menjelaskan konsep hubungan antara sistem dengan dunia luar. *Use case diagram* dapat dilihat pada gambar II.2 seperti berikut.



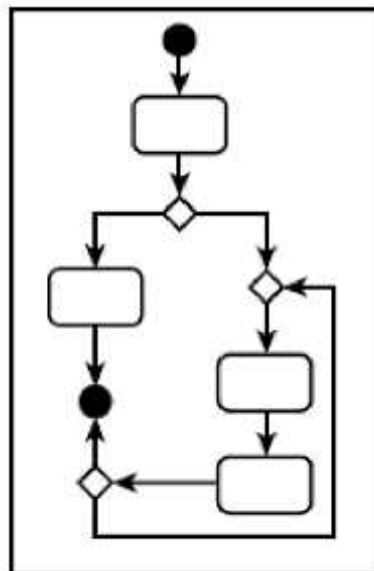
**Gambar II.2 Use Case Diagram**  
(Sumber : Mulyani ; 2016)

- 2) *Use case narrative* yaitu deskripsi yang menjelaskan use case diagram. Pada *use case diagram* sistem hanya digambarkan secara sederhana menggunakan simbol *use case* yang berhubungan (*relationship*) dengan aktor, sehingga terkadang diperlukan deskripsi yang menjelaskan dan proses tersebut. *Use case narrative* dapat dilihat pada gambar II.3 seperti berikut.



**Gambar II.3** *Use Case Narrative*  
(Sumber : Mulyani ; 2016)

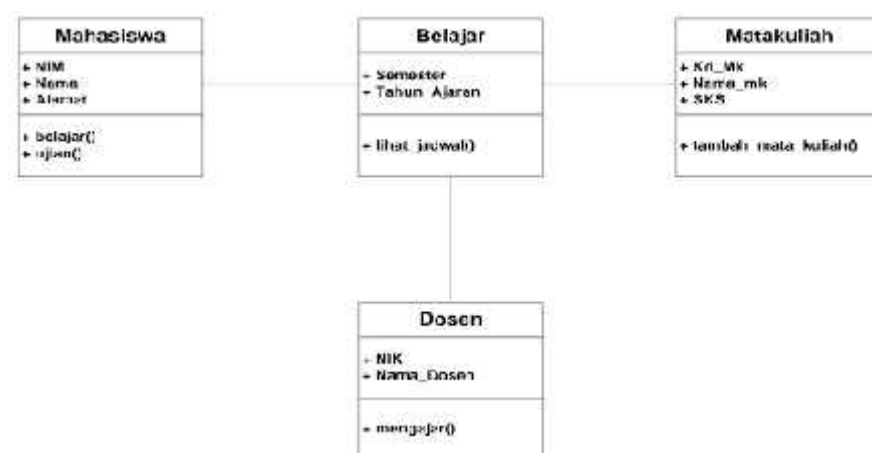
- 3) *Use case scenario* yaitu pemecahan kemungkinan logika pada *use case diagram*. *Use case scenario* dapat dilihat pada gambar II.4 seperti berikut.



**Gambar II.4** *Use Case Scendario*  
(Sumber : Mulyani ; 2016)

## 2. Class Diagram

*Class diagram* adalah diagram yang digunakan untuk merepresentasikan kelas, komponen-komponen kelas dan hubungan antara masing-masing kelas. Selain itu class diagram mendeskripsikan jenis-jenis objek dalam sistem dan berbagai macam hubungan statis yang terdapat diantara mereka. *Class diagram* juga menunjukkan *property* dan operasi sebuah kelas serta batasan-batasan yang terdapat dalam hubungan-hubungan objek tersebut. *UML* menggunakan istilah fitur sebagai istilah umum yang meliputi *property* dan operasi sebuah kelas. *Class diagram* dapat dilihat pada gambar II.5 seperti berikut.

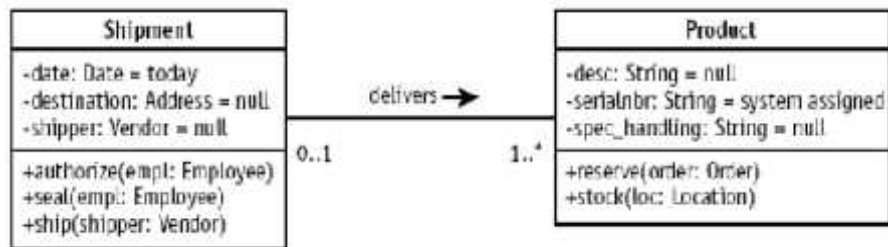


**Gambar II.5 Class Diagram**  
(Sumber : Mulyani ; 2016)

## 3. Object Diagram

*Object diagram* adalah diagram yang digunakan untuk menggambarkan dan merepresentasikan objek dan hubungan antar objek tersebut, selain itu *object diagram* juga dapat digunakan sebagai diagram untuk menunjukan

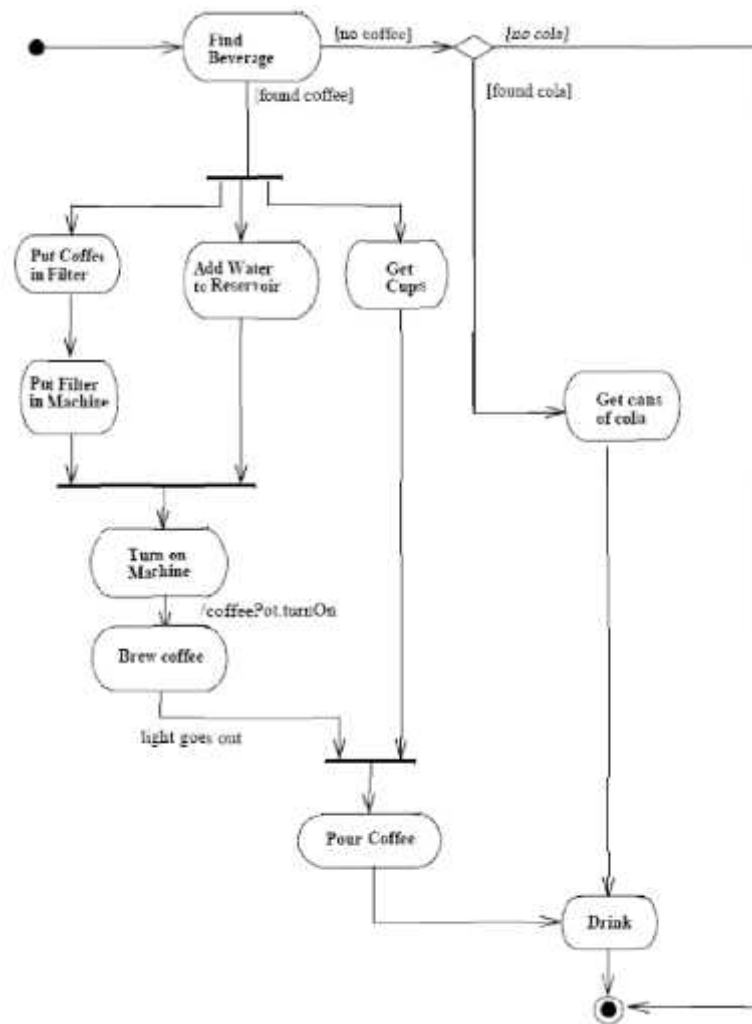
sebuah konfigurasi contoh dan sebuah objek. *Object diagram* dapat dilihat pada gambar II.6 seperti berikut.



**Gambar II.6 Object Diagram**  
(Sumber : Mulyani ; 2016)

#### 4. Activity Diagram

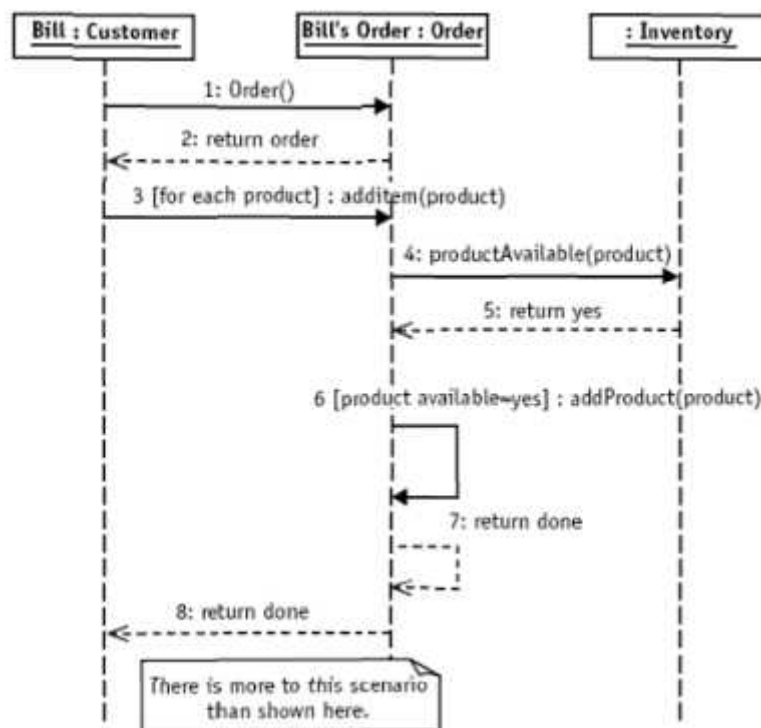
Jika sebelumnya Anda sudah pernah menggunakan *flowchart*, maka Anda akan mudah dalam memahami dan mempelajari *activity diagram*, karena *diagram* ini sangat mirip sekali dengan *fiowchart*, perbedaannya adalah *activity diagram* memiliki kemampuan untuk melakukan percabangan aktivitas, selain itu *activity diagram* juga memungkinkan pemisahan aktivitas antar aktor. *Activity diagram* adalah diagram *UML* yang digunakan untuk menggambarkan alur aktivitas dari satu proses. *Activity diagram* memungkinkan siapapun yang melakukan proses untuk memilih urutan dalam melakukannya, dengan kata lain diagram hanya menyebutkan aturan-aturan rangkaian dasar yang harus kita ikuti. Hal ini penting untuk pemodelan bisnis karena proses-proses sering muncul secara parallel. Ini juga berguna pada algoritma yang bersamaan, dimana urutan-urutan independen dapat melakukan hal-hal secara parallel. *Acitivity diagram* dapat dilihat pada gambar II.7 seperti berikut.



**Gambar II.7 Activity Diagram**  
(Sumber : Mulyani ; 2016)

### 5. Sequence Diagram

*Sequence diagram* adalah diagram yang menggambarkan interaksi antar objek. *Sequence diagram* secara khusus menjabarkan *behavior* sebuah skenario tunggal. Diagram tersebut menunjukkan sejumlah objek contoh dan pesan-pesan yang melewati objek ini dalam sebuah *use case*. *Sequence diagram* dapat dilihat pada gambar II.8 seperti berikut.



**Gambar II.8 Sequence Diagram**  
(Sumber : Mulyani ; 2016)

## II.9 Normalisasi (Normalization)

Normalisasi adalah salah satu cara untuk meminimalisir pengulangan data (*data redundancy*), normalisasi akan diperlukan jika ada indikasi bahwa tabel yang kita buat tidak baik (terjadi pengulangan informasi, potensi inkonsistensi data pada operasi perubahan, tersembunyinya informasi tertentu dan lain sebagainya) dan diperlukan supaya jika tabel-tabel yang didekomposisi kita gabung kembali dapat menghasilkan tabel awal sebelum didekomposisi, sehingga diperoleh tabel yang baik. Hasil dari normalisasi adalah himpunan-himpunan data (tabel-tabel) dalam bentuk normal (normal form). (Mulyani, 2016 : 132)

Beberapa kegunaan normalisasi adalah (Mulyani, 2016 : 132):

1. Meminimalisir pengulangan data (*data redundancy*)
2. Memudahkan identifikasi *entity* objek.

Beberapa bentuk normal yaitu (Mulyani, 2016 : 132):

1. Bentuk Normal I (*First Normal Form / 1-NF*)

Suatu relasi memenuhi *1-NF* jika dan hanya jika setiap *attribute* dan relasi tersebut hanya memiliki nilai tunggal dalam 1 (satu) baris *record* (memisahkan group berulang).

2. Bentuk Normal II (*Second Normal Form/2-NF*)

Suatu relasi memenuhi *2-NF* jika dan hanya jika memenuhi *1-NF* dan Setiap *attribute* yang bukan kunci utama tergantung secara fungsional terhadap semua attribute kunci dan bukan hanya sebagian attribute (menghilangkan ketergantungan fungsional pada sebagian/salah satu key).

3. Bentuk Normal III (*Third Normal Form/3-NF*)

Suatu relasi memenuhi *3-NF* jika dan hanya jika memenuhi *2-NF* dan setiap *attribute* bukan kunci tidak tergantung secara fungsional kepada *attribute* bukan kunci yang lain dalam relasi tersebut (menghilangkan ketergantungan transitif pada yang bukan *key*).

4. Boyce-Codd Normal Form (BCNF)

Suatu relasi memenuhi BCNF jika untuk setiap functional dependency terhadap setiap atribut atau gabungan atribut dalam bentuk :  $X \rightarrow Y$  maka X adalah super key. Tabel tersebut harus di-dekomposisi berdasarkan functional dependency yang ada, sehingga X menjadi super key dari tabel-

tabel hasil dekomposisi. Setiap tabel dalam BCNF merupakan 3NF. Akan tetapi setiap 3NF belum tentu termasuk BCNF. Perbedaannya, untuk functional dependency  $X \rightarrow A$ , BCNF tidak membolehkan A sebagai bagian dari primary key.

#### 5. Bentuk Normal IV (*Fourth Normal Form/4-NF*)

Suatu relasi memenuhi *4-NF* jika dan hanya jika memenuhi *BCNF* dan tabel tersebut tidak boleh memiliki lebih dari sebuah *multivalued attribute*.

Untuk setiap *multivalued attribute (MVD)* juga harus merupakan *functional dependencies*.

Beberapa key dalam Normalisasi (Mulyani, 2016 : 132):

1. *Superkey* adalah sejumlah *attribute entity* yang dapat digunakan untuk mengidentifikasi objek secara unik.
2. *Candidate key* adalah *superkey* dengan jumlah attribute minimal dan dapat berdiri sendiri.
3. *Primary key* adalah *superkey* yang dipilih oleh desainer atau database administrator.
4. *Foreign key* adalah *attribute* disuatu relasi (*tabel*) yang menjadi *primary key* di relasi (*label*) lain.

### II.10 *Basis Data (Database)*

*Database* adalah kumpulan dari semua data yang diperlukan oleh sistem.

Dengan menggunakan database, beberapa aplikasi berbeda bisa saling

terintegrasi, misalnya aplikasi keuangan, aplikasi kepegawaian dengan penggajian atau aplikasi persediaan. (Mulyani, 2016 : 148).

### **II.10.1 DBMS (*Database Management System*)**

*Database Management System* adalah sistem *software* yang memungkinkan pengguna untuk mendefinisikan, membuat, memelihara, dan kontrol akses ke database. (Mulyani, 2016 : 170).

*DBMS* adalah *software* yang berinteraksi dengan program aplikasi dan pengguna database Biasanya *DBMS* menyediakan fasilitas sebagai berikut :

1. *DDL (Data Definition Language)*

*DDL* memungkinkan pengguna untuk menentukan tipe data dan struktur dan kendala pada data yang akan disimpan dalam *database*.

2. *DML(Data Manipulation Language)*

Ini memungkinkan pengguna untuk memasukkan, update, menghapus dan mengambil data dari *database* biasanya meskipun memanipulasi data bahasa (*DML*).

3. Memberikan akses kontrol ke *database* :

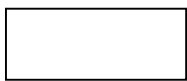

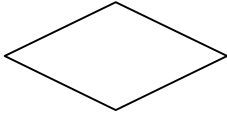

- 1) Keamanan sistem: yang mencegah pengguna yang tidak berhak mengakses database.
- 2) Integritas system: yang menjaga konsistensi data yang tersimpan.
- 3) *Concurrency control system*: yang memungkinkan berbagi akses database.

- 4) Pemulihan sistem *control*: yang mengembalikan *database* ke keadaan yang konsisten sebelumnya setelah perangkat keras atau kegagalan *software*.
- 5) *User*-diakses katalog, yang berisi deskripsi dari data dalam *database*.

### II.10.2 Entity Relationship Diagram (ERD)

*Entity Relationship Diagram (ERD)* digunakan untuk menggambarkan hubungan antara penyimpanan atau data storage yang dapat pada *DFD*, *ERD* menggunakan sejumlah notasi untuk menggambarkan struktur dan hubungan antar data. (Mulyani, 2016 : 100). Notasi atau simbol-simbol yang digunakan pada *ERD* dapat dilihat sebagai berikut :

**Tabel II.1 Simbol ERD**

Simbol	Nama	Keterangan
	Entitas	Jenis entitas dapat berupa suatu elemen lingkungan, sumber daya atau transaksi yang field-fieldnya dipergunakan dalam aplikasi program
	Atribut	Atribut adalah karakteristik dari sebuah entitas
	Relasi	Menunjukkan nama relasi antar satu entitas dengan entitas lainnya.
	Garis Relasi	Menunjukkan hubungan (keterkaitan) antar entitas satu dengan lainnya.

(Sumber : Mulyani ; 2016)