

## BAB II

### LANDASAN TEORI

#### II.1 Sistem

Sistem merupakan serangkaian bagian yang saling tergantung dan bekerja sama untuk mencapai tujuan tertentu. Suatu sistem pasti tersusun dari sub-sub sistem yang lebih kecil yang juga saling tergantung dan bekerja sama untuk mencapai tujuan. Tujuan dasar suatu sistem tergantung pada jenis sistem itu sendiri. Termasuk juga sistem buatan manusia seperti sistem yang terdapat di sekolah, organisasi bisnis, atau instansi pemerintah juga mempunyai tujuan yang berbeda-beda (Anastasia Diana dan Lilis Setiawati: 2011;3).

Menurut Jogiyanto, sistem adalah kumpulan elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu (2005). Sistem adalah kumpulan himpunan dari unsur atau variabel-variabel yang saling terkait, saling berinteraksi, dan saling tergantung satu sama lain untuk mencapai tujuan. Karakteristik suatu sistem :

1. Komponen atau elemen (*Components*), suatu sistem terdiri dari komponen-komponen yang saling berinteraksi, yang artinya saling bekerjasama membentuk satu kesatuan.
2. Batas sistem (*Boundary*), merupakan daerah yang membatasi antara sistem yang satu dengan sistem yang lainnya atau dengan lingkungan luarnya.
3. Lingkungan luar sistem (*Environment*), segala sesuatu diluar batas sistem yang mempengaruhi operasi suatu sistem.

4. Penghubung sistem (*Interface*), suatu media atau penghubung antara satu subsistem dengan subsistem lainnya yang membentuk satu kesatuan, sehingga sumber-sumber daya mengalir dari subsistem yang satu ke subsistem yang lainnya.
5. Masukan (*Input*), energy atau sesuatu yang dimasukkan ke dalam suatu sistem yang dapat berupa masukan yang dapat diproses untuk menghasilkan suatu luaran.
6. Luaran (*Output*), hasil dari energi yang diolah dan diklasifikasikan menjadi luaran yang berguna dan tujuan akhir dari sistem.
7. Pengolah (*Process*), suatu sistem mempunyai bagian pengolah yang akan mengubah *input* menjadi *output*.
8. Sasaran. Sasaran dari sistem sangat menentukan sekali masukan yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem (Hamim Tohari : 2014 ; 2-3).

## **II.2. Sistem Informasi**

Sistem informasi, yang kadang kala disebut sebagai sistem pemrosesan data, merupakan sistem buatan manusia yang biasanya terdiri dari sekumpulan komponen, baik manual ataupun berbasis komputer yang terintegrasi untuk mengumpulkan, menyimpan, dan mengelola data serta menyediakan informasi kepada pihak-pihak yang berkepentingan sebagai pemakai informasi tersebut.

Input dalam sistem informasi adalah data-data yang relevan untuk menghasilkan informasi yang diinginkan. Proses adalah langkah-langkah yang

perlu dilakukan untuk mengolah data menjadi informasi. Output adalah berupa informasi yang merupakan hasil dari pemrosesan data (Anastasia Diana dan Lilis Setiawati: 2011;4).

Informasi merupakan aset penting bagi suatu institusi atau organisasi. Oleh karena itu, informasi harus berkualitas, dijaga, dan dipelihara dengan baik. Sedangkan kualitas dari suatu informasi tergantung 3 hal:

1. Akurat, informasi harus bebas dari kesalahan-kesalahan.
2. Tepat waktu, informasi yang datang pada penerima tidak boleh terlambat.  
Informasi yang sudah tidak berlaku tidak akan mempunyai nilai lagi karena informasi merupakan landasan diadakannya pengambilan keputusan.
3. Relevan, informasi harus mempunyai manfaat bagi pemakainya. Relevansi informasi untuk tiap-tiap orang berbeda (Hamim Tohari: 2014;7-8).

### **II.3. Sistem Informasi Akuntansi**

Sistem informasi akuntansi adalah sistem informasi yang menyajikan informasi yang dipakai oleh fungsi akuntansi. Sistem ini mencakup semua transaksi yang berhubungan keuangan di sebuah perusahaan atau organisasi (Hamim Tohari: 2014;8).

Sistem informasi akuntansi adalah sistem yang bertujuan untuk mengumpulkan dan memproses data serta melaporkan informasi yang berkaitan dengan transaksi keuangan. Lingkup sistem informasi akuntansi dapat dijelaskan dari manfaat yang didapat dari informasi akuntansi. Manfaat atau tujuan sistem informasi akuntansi tersebut adalah sebagai berikut :

1. Mengamankan harta/kekayaan perusahaan, meliputi kas perusahaan, persediaan barang dagangan, dan asset tetap perusahaan.
2. Menghasilkan beragam informasi untuk pengambilan keputusan. Misalnya pengelola toko swalayan memerlukan informasi mengenai barang apa saja yang diminati oleh konsumen. Membeli barang yang kurang laku berarti kas akan terjebak dalam persediaan dan berarti kehilangan kesempatan untuk membeli barang dagangan yang laku.
3. Menghasilkan informasi untuk pihak eksternal. Setiap pengelola usaha memiliki kewajiban untuk membayar pajak.
4. Menghasilkan informasi untuk penilaian kinerja karyawan atau divisi. Sistem informasi dapat juga dimanfaatkan untuk penilaian kinerja karyawan atau divisi.
5. Menyediakan data masa lalu untuk kepentingan audit (pemeriksaan).
6. Menghasilkan informasi untuk penyusunan dan evaluasi anggaran perusahaan. Anggaran merupakan alat yang sering digunakan perusahaan untuk mengendalikan pengeluaran kas.
7. Menghasilkan informasi yang diperlukan dalam kegiatan perencanaan dan pengendalian. Selain untuk membandingkan informasi yang berkaitan dengan anggaran dan biaya standar dengan kenyataan seperti yang telah dikemukakan.

Sistem informasi akuntansi memiliki peranan yang penting dalam proses bisnis karena sistem informasi akuntansi mengidentifikasi, mengukur dan mencatat proses bisnis tersebut dalam suatu model yang sedemikian rupa sehingga

informasi yang dihasilkan dapat dimanfaatkan oleh pihak yang berkepentingan (Anastasia Diana dan lilis setiawati: 2011;4-7).

#### **II.4. Penyusutan Kendaraan**

Dr. Erhans Anggawirya “penyusutan adalah alokasi jumlah suatu aktiva yang dapat disusutkan sepanjang masa manfaat yang diestimasi (PSAK No. 17 tahun 2002). Jumlah aktiva yang dapat disusutkan adalah biaya perolehan dikurangi nilai residu. Jika nilai residu tidak ada, maka jumlah aktiva yang dapat disusutkan sebesar biaya perolehan (Nisa UI Kholqiah & Nining R: 2011;40)

Aktiva tetap kecuali tanah atau hak atas tanah pada waktu digunakan dalam operasi perusahaan yang dimaksudkan untuk memperoleh laba, kegunaannya akan semakin menurun. Penurunan kegunaan aktiva tetap tersebut mengakibatkan nilainya harus disusutkan. Proses itu dinamakan penyusutan untuk aktiva berwujud yang dapat diganti. Proses penyusutan ini penekanan utamanya adalah pada pengalokasian biaya dari *cost* aktiva tetap ke biaya periode untuk ditandingkan dengan pendapatan yang dilaporkan pada masing-masing periode selama digunakan aktiva tetap tersebut (Arif Rohman & Nining R: 2010;3-4).

Depresiasi atau penyusutan adalah pengalokasian harga perolehan aktiva tetap, seperti kendaraan, gedung, mesin produksi ataupun peralatan selama umur ekonomis aktiva tersebut. Umur ekonomis adalah berapa lama aktiva tersebut bermanfaat bagi perusahaan secara efisien. Jadi, umur ekonomis tidak sama dengan umur aktiva (Anastasia Diana dan Lilis Setiawati: 2011;24).

Menurut Hery dan Widyawati Lekok (2011;22), penyusutan bukanlah proses dimana perusahaan mengakumulasikan dana (kas) untuk mengganti aktiva tetapnya, contohnya kendaraan. Penyusutan juga bukan cara menghitung nilai yang berlaku saat ini atas aktiva tetap. Penyusutan adalah alokasi secara periodik dan sistematis dari harga perolehan aktiva (seperti kendaraan, bangunan, dan peralatan) selama periode-periode berbeda yang memperoleh manfaat dari penggunaan aktiva bersangkutan (kendaraan). Akumulasi penyusutan adalah bukan sebuah dana pengganti aktiva, melainkan jumlah harga perolehan aktiva yang telah dibebankan (melalui pemakaian) dalam periode-periode sebelumnya. Nilai buku aktiva (harga perolehan, yang merupakan biaya historis, dikurang dengan akumulasi penyusutan) adalah harga perolehan aktiva (kendaraan) yang tersisa yang akan dialokasikan untuk pemakaian di periode yang akan datang, dan bukan merupakan estimasi atas nilai aktiva tetap saat ini.

Penyusutan umumnya terjadi ketika aktiva tetap (kendaraan) telah digunakan dan merupakan beban bagi periode dimana aktiva dimanfaatkan. Penyusutan dilakukan karena masa manfaat dan potensi aktiva yang dimiliki semakin berkurang. Pengurangan nilai aktiva (kendaraan) tersebut dibebankan secara berangsur-angsur atau proporsional ke masing-masing periode yang menerima manfaat.

Jadi, beban penyusutan adalah pengakuan atas penggunaan manfaat potensial dari suatu aktiva (kendaraan). Beban penyusutan merupakan beban yang tidak memerlukan pengeluaran uang kas (*non cash outlay expense*). Alokasi harga perolehan aktiva tetap dilakukan dengan cara mendebet akun beban

penyusutan dan mengkredit akun akumulasi penyusutan. Bentuk umum dari ayat jurnal yang digunakan untuk mengakui beban penyusutan adalah:

Beban penyusutan	xxx	
	Akumulasi penyusutan	xxx

(Hery dan Widyawati Lekok: 2011;22)

#### **II.4.1. Ketentuan Umum Penyusutan Aktiva Tetap**

Melalui ketentuan Pasal 9 ayat (2), UU PPh secara tegas menyatakan bahwa pengeluaran untuk mendapatkan, menagih dan memelihara penghasilan yang mempunyai masa manfaat lebih dari 1 (satu) tahun tidak dibolehkan untuk dibebankan sekaligus, melainkan dibebankan melalui penyusutan atau amortisasi sebagaimana dimaksud dalam Pasal 11 atau Pasal 11A.

*Pasal 11 UU PPh secara umum berisi ketentuan mengenai penyusutan untuk harta berwujud sedangkan Pasal 11A UU PPh berisi ketentuan mengenai amortisasi atas pengeluaran untuk memperoleh harta tak berwujud termasuk HGB, HGU, Hak Pakai, Goodwill, dan harta atau asset tak berwujud lainnya.*

Secara khusus, otoritas pajak telah menetapkan beberapa ketentuan khusus yang diatur dalam peraturan-peraturan berikut (yang masih berlaku sampai saat artikel ini ditulis):

- a. Pasal 11 dan Pasal 11A UU PPh;
- b. Peraturan Menteri Keuangan (PMK) Nomor 96/PMK.03/2009; dan
- c. PMK Nomor 249/PMK.03/2008 stdd PMK Nomor 126/PMK.03/2012

## 1. Pengelompokan Harta Berwujud

PPh mengelompokkan harta berwujud menjadi dua, yaitu harta berwujud berupa bangunan dan bukan bangunan. Kelompok harta berwujud yang bukan bangunan dikelompokkan menjadi 4 kelompok, yaitu kelompok 1 sampai dengan kelompok 4 yang diklasifikasikan berdasarkan masa manfaatnya. Pengelompokan tersebut dapat dilihat pada Peraturan Menteri Keuangan nomor 96/PMK.03/2009. Dan untuk jenis-jenis harta berwujud yang tidak terdapat pada lampiran PMK tersebut, untuk kepentingan penyusutan digunakan masa manfaat dalam kelompok 3, kecuali jika Wajib Pajak dapat menunjukkan bahwa masa manfaat sesungguhnya dari harta tersebut tidak dapat dimasukkan sebagai harta dengan masa manfaat kelompok 3, maka Wajib Pajak harus mengajukan permohonan untuk penetapan kelompok harta berwujud tersebut sesuai dengan masa manfaat yang sesungguhnya kepada DJP melalui Kepala Kanwil DJP yang membawahi KPP tempat WP terdaftar, yang tata caranya diatur dalam Peraturan Direktur Jenderal Pajak nomor PER-55/PJ/2009.

### **Kelompok 2 (Masa manfaat 8 tahun)**

No	Jenis Usaha	Jenis Harta
1	Semua jenis usaha	a. Mebel dan peralatan dari logam termasuk meja, bangku, kursi, lemari dan sejenisnya yang bukan merupakan bagian dari bangunan. Alat pengatur udara seperti AC, kipas angin dan sejenisnya. b. Mobil, bus, truk, speed boat dan sejenisnya.

- c. Container dan sejenisnya.
- 2
  - a. Mesin pertanian/perkebunan seperti traktor dan mesin bajak, penggaruk, penanaman, penebar benih dan sejenisnya.
  - b. Mesin yang mengolah atau menghasilkan atau memproduksi bahan atau barang pertanian, perkebunan, peternakan dan perikanan.
      - a. Mesin yang mengolah produk asal binatang, unggas dan perikanan, misalnya pabrik susu, pengalengan ikan.
      - b. Mesin yang mengolah produk nabati, misalnya mesin minyak kelapa, margarin, penggilingan kopi, kembang gula, mesin pengolah biji-bijian seperti penggilingan beras, gandum, tapioka.
    - c. Mesin yang menghasilkan/memproduksi minuman dan bahan-bahan minuman segala jenis.
    - d. Mesin yang menghasilkan/memproduksi bahan-bahan makanan dan makanan segala jenis.
- 3
  - Mesin yang menghasilkan/memproduksi mesin ringan (misalnya mesin jahit, pompa air).
- 4
  - Industri makanan dan minuman
- Industri mesin

- 5           Perkayuan,  
              kehutanan
- 6           Konstruksi
- 7           Transportasi dan  
              Pergudangan
- a. Mesin dan peralatan penebangan kayu.
  - b. Mesin yang mengolah atau menghasilkan atau memproduksi bahan atau barang kehutanan.  
  
Peralatan yang dipergunakan seperti truk berat, dump truck, crane buldozer dan sejenisnya.
  - a. Truk kerja untuk pengangkutan dan bongkar muat, truk peron, truck ngangkang, dan sejenisnya;
  - b. Kapal penumpang, kapal barang, kapal khusus dibuat untuk pengangkutan barang tertentu (misalnya gandum, batu – batuan, biji tambang dan sebagainya) termasuk kapal pendingin, kapal tangki, kapal penangkap ikan dan sejenisnya, yang mempunyai berat sampai dengan 100 DWT;
  - c. Kapal yang dibuat khusus untuk menghela atau mendorong kapal-kapal suar, kapal pemadam kebakaran, kapal keruk, keran terapung dan sejenisnya yang mempunyai berat sampai dengan 100 DWT;
  - d. Perahu layar pakai atau tanpa motor yang mempunyai berat sampai dengan 250 DWT;

- e. Kapal balon.
- a. Perangkat pesawat telepon.
- b. Pesawat telegraf termasuk pesawat pengiriman dan penerimaan radio telegraf dan radio telepon.
- 8 Telekomunikasi
- Jasa persewaan
- 9 peralatan tambat air dalam *Spooling Machines, Metocean Data Collector*
- Jasa
- 10 telekomunikasi seluler *Mobile Switching Center, Home Location Register, Visitor Location Register. Authentication Centre, Equipment Identity Register, Intelligent Network Service Control Point, intelligent Network Service Managemen Point, Radio Base Station, Transceiver Unit, Terminal SDH/Mini Link, Antena. (Trio Mandala Putra : 2013 ; 192-197)*

Golongan aset tetap berwujud dan tarif penyusutan menurut UU PPh :

**Tabel II.1. Golongan Aset Tetap Berwujud dan Tarif Penyusutan**

Kelompok Harta Berwujud	Masa Manfaat	Tarif penyusutan sebagaimana dimaksud dalam	
		Ayat (1)	Ayat (2)
I. Bukan Bangunan			
Kelompok 1	4 Tahun	25%	50%
Kelompok 2	8 Tahun	12,5%	25%
Kelompok 3	16 Tahun	6,25%	12,5%
Kelompok 4	20 Tahun	5%	10%
II. Bangunan			
Permanen	20 Tahun	5%	
Tidak Permanen	10 Tahun	10%	

( Sumber : Trio Mandala Putra : 2013 )

#### II.4.2. Faktor-Faktor yang Mempengaruhi Beban Penyusutan

Untuk memperoleh besarnya beban penyusutan periodik secara tepat dari pemakaian suatu aset, ada tiga faktor yang perlu dipertimbangkan, yaitu :

a. Nilai perolehan aset (*asset cost*)

Nilai perolehan suatu aset mencakup seluruh pengeluaran yang terkait dengan perolehannya dan persiapannya sampai aset dapat digunakan. Jadi, disamping harga beli, pengeluaran-pengeluaran lain yang diperlukan untuk mendapatkan dan mempersiapkan aset harus disertakan sebagai harga perolehan. Nilai perolehan aset umumnya mencerminkan nilai pasar pada saat aset diperoleh.

b. Nilai residu/sisa (*residual or salvage value*)

Nilai sisa merupakan estimasi nilai realisasi pada saat aset tidak dipakai lagi. Dengan kata lain, nilai sisa ini mencerminkan nilai estimasi dimana aset dapat dijual kembali ketika aset tetap tersebut dihentikan dari pemakaiannya (pada saat estimasi masa manfaat aset berakhir). Besarnya estimasi nilai sisa sangat tergantung pada kebijakan manajemen mengenai penghentian aset tetap, dan juga tergantung pada kondisi pasar serta faktor lainnya. Nilai sisa sifatnya adalah subyektif, dimana sangat tergantung pada kebijakan manajemen dari masing-masing perusahaan.

c. Umur ekonomis (*economic life*)

Dalam menghitung besarnya beban penyusutan, umur ekonomis dapat diartikan sebagai suatu periode atau umur fisik dimana perusahaan dapat memanfaatkan aset tetapnya (masa manfaat) dan dapat juga berarti sebagai jumlah unit produksi (output) atau jumlah jam operasional (jasa) yang diharapkan diperoleh dari aset. Faktor-faktor fisik yang membatasi umur ekonomis suatu aset mencakup pemakaian, penurunan nilai (berhubungan dengan berlalunya waktu, dimana suatu aset tetap baik digunakan atau tidak digunakan akan mengalami penurunan nilai), dan kerusakan (penyebabnya dapat berupa kebakaran, banjir, gempa bumi atau kecelakaan yang cenderung mengurangi atau mengakhiri usia manfaat suatu aset), sedangkan faktor fungsional yang membatasi umur aset adalah keusangan (*obsolescence*) (Hery: 2014;276-278).

## II.5. Metode Jam Jasa

Metode jam jasa (*service hours method*) digunakan untuk mengalokasikan beban penyusutan berdasar propors penggunaan aktiva yang sebenarnya (Arif Rohman dan Nining R: 2010;5).

Teori yang mendasari metode ini adalah bahwa pembelian suatu aktiva menunjukkan pembelian sejumlah jam jasa langsung. Dalam menghitung besarnya beban penyusutan, metode ini membutuhkan estimasi umur aktiva berupa jumlah jam jasa yang dapat diberikan oleh aktiva bersangkutan. Harga perolehan yang dapat disusutkan (harga perolehan dikurangi dengan nilai estimasi nilai residu) dibagi dengan estimasi total jam jasa, menghasilkan besarnya tarif penyusutan untuk setiap jam pemakaian aktiva.

$$\text{Penyusutan} = \frac{\text{Harga Perolehan} - \text{Nilai Residu}}{\text{Estimasi Total Jam Jasa}}$$

**Gambar II.1. Rumus Penyusutan**  
(Sumber : Hery ; 2014)

Pemakaian aktiva sepanjang periode (jumlah jam jasanya) dikalikan dengan tarif penyusutan tersebut akan menghasilkan besarnya beban penyusutan periodik. Besarnya beban penyusutan ini akan berfluktuasi setiap periodenya tergantung pada jumlah kontribusi jam jasa yang diberikan oleh aktiva bersangkutan.

Sebagai contoh, asumsi bahwa pada akhir bulan Maret 2010 dibeli sebuah aktiva tetap dengan harga perolehan sebesar Rp. 100.000.000. berdasarkan estimasi manajemen, aktiva tetap ini diperkirakan dapat beroperasi selama 25.000 jam dengan nilai sisa sebesar Rp. 5.000.000. Dengan contoh tersebut, dan apabila

metode jam jasa diterapkan, maka besarnya tarif penyusutan untuk setiap jam pemakaian aktiva adalah :

$$(\text{Rp. } 100.000.000 - \text{Rp. } 5.000.000) : 25.000 \text{ jam} = \text{Rp. } 3.800 \text{ per jam.}$$

Jika sepanjang tahun 2010 aktiva tersebut telah dipakai selama 2.400 jam, maka besarnya beban penyusutan untuk pemakaian tahun 2010 akan menjadi Rp. 3.800/jam x 4.200 jam = Rp. 15. 960.000 (Hery, 2014;289)

Beban penyusutan	Rp. 15. 960.000
Akumulasi penyusutan	Rp. 15. 960.000

## II.6. Aplikasi

Menurut Panji M. Sudarmo (2006 : 21) aplikasi yaitu sejenis tugas atau pekerjaan yang dilakukan suatu program atau system computer misalnya perancangan teknik, system pemesanan tiket pesawat terbang, administrasi keuangan dan sebagainya (Arif Rohman dan Nining R: 2010;4).

### II.6.1. Adobe Dreamweaver

Dalam membuat sebuah website memerlukan program aplikasi yang andal sekaligus mudah dalam membangun web tersebut. Salah satu program aplikasi yang sekarang ini banyak digunakan adalah *Adobe Dreamweaver*.

*Adobe Dreamweaver* adalah aplikasi yang memberikan tampilan yang lebih baik dan tentu saja semakin mudah dalam penggunaannya, aplikasi ini mengintegrasikan beragam fitur untuk memenuhi kebutuhan pengembangan website, termasuk pembuatan halaman web dan pengelolaannya. *Adobe*

*Dreamweaver* menyertakan banyak *tool* yang berkaitan dengan pengkodean seperti HTML, CSS, XML dan pemrograman *Client Side*, yaitu *JavaScript* dengan penggunaan yang sangat mudah dan *user Friendly*. Aplikasi ini juga mendukung pemrograman *Script Server Side* seperti PHP, *Active Server Page* (ASP), ASP.NET, ASP *JavaScript*, ASP *VBScript*, *ColdFusion*, dan *Java Server Page* (JSP) (Andi & Wahana Komputer, 2011;2).

Terdapat beberapa macam tipe file untuk format halaman web yang dapat Anda pilih, seperti:

1. HTML (*Hypertext Markup Language*)

HTML merupakan dasar untuk pembuatan desain web. *File* HTML berisi instruksi tertentu yang dapat memberikan suatu format dokumen yang akan ditampilkan pada *World Wide Web*.

2. *ColdFusion*

Merupakan bahasa scripting yang digunakan oleh Adobe *ColdFusion*, *BlueDragon*, dan sebagainya untuk *scripting server-side*.

3. PHP (*Hypertext Preprocessor*)

Adalah sebuah bahasa pemrograman yang umum dipakai untuk *scripting server-side*. PHP biasanya terpasang pada HTML, dengan bahasa pemrograman ini dapat dibuat suatu web yang dinamis.

4. ASP *VBScript* (*ASP Visual Basic Script*)

*VBScript* merupakan bahasa *scripting* turunan dari bahasa pemrograman *Visual Basic* yang dapat digunakan untuk membuat sebuah aplikasi HTML (yang memiliki ekstensi .HTA).

#### 5. XSLT (*Extensible Stylesheet Language Transformations*)

Adalah bahasa pemrograman berdasar XML yang digunakan untuk transformasi dokumen XML menjadi dokumen XML atau format lainnya.

#### 6. CSS (*Cascading Style Sheet*)

Adalah bahasa *Stylesheet* yang digunakan untuk mengatur tampilan halaman web dan ditulis dengan HTML atau XHTML. CSS dapat juga digunakan untuk semua jenis dokumen XML termasuk SVG dan XUL.

#### 7. JavaScript

Adalah bahasa *scripting* yang mempunyai kesamaan dengan penggunaan sintaks bahasa pemrograman C. *Script* ini umum digunakan untuk pengembangan *web client-side*.

#### 8. XML (*Extensible Markup Language*)

Menggunakan markup tags seperti halnya HTML, namun penggunaannya tidak terbatas pada halaman web saja.

#### 9. Dreamweaver Site

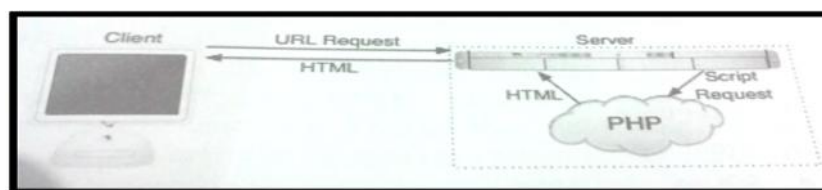
Berfungsi untuk membuat websit baru dengan program *Dreamweaver* (Andi & Wahana Komputer: 2011;9).

### **II.6.2. Bahasa Pemrograman PHP**

PHP merupakan bahasa skrip yang ditanam dalam HTML. Ini berarti bahwa anda dapat menggabungkan kode PHP dan HTML dalam *file* yang sama.

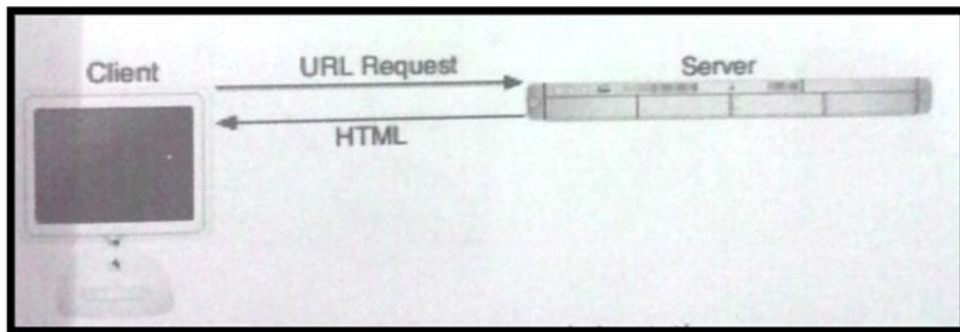
PHP (*Hypertext Preprocessor*) merupakan salah satu bahasa pemrograman web *server-side* yang populer dan banyak digunakan sampai saat ini. PHP dapat diperoleh secara *free* dengan mendownload di situs resmi <http://www.php.net/>, atau menggunakan paket instan yang telah terintegrasi dengan berbagai aplikasi untuk kebutuhan server dan database. Selain sebagai bahasa pemrograman web yang dapat dieksekusi sendiri, PHP juga dikenal sebagai *embedded language*. Artinya, anda dapat memasukkan kode PHP ke dalam bahasa html yang sebelumnya dikenal sebagai bahasa pembentuk halaman website.

PHP dieksekusi di sisi server, sehingga PHP dapat mengambil data, mengolah, dan mengirimkan data dari database server ke halaman browser. Isi halaman web dapat diubah sesuai keinginan menggunakan konten dalam database. Konten halaman web juga dapat berasal dari pengunjung, lalu diolah di database. Konten tersebut dapat ditampilkan pada halaman web. Proses ini disebut website dinamis. Berikut ini gambar perbedaan proses yang terjadi pada website yang dinamis dan website statis.



**Gambar II.2. Proses Website Dinamis**  
(Sumber : Eko Priyo Utomo ; 2014)

Pada gambar di atas, ketika klien meminta *request* ke server, maka akan dilakukan cek keberadaan skrip PHP di server. Setelah proses selesai, skrip dikirimkan kembali ke klien dalam format HTML dan tampil pada layar monitor.



**Gambar II.3. Proses Website Statis**  
(Sumber : Eko Priyo Utomo ; 2014)

Pada website statis, *request* yang diminta langsung dikembalikan ke klien dalam format HTML tanpa melakukan cek keberadaan skrip.

PHP dapat mendukung berbagai macam database, antara lain MySQL, Oracle, dBase, Solid, dan lain-lain (Eko Priyo Utomo: 2014;2-5).

## II.7. MySQL

MySQL merupakan salah satu database yang banyak digunakan oleh para pengguna komputer. MySQL menggunakan bahasa SQL (*structured query language*) untuk berinteraksi dengan database. Setiap database menggunakan bahasa SQL dalam pengoperasiannya. Pada bahasa SQL, perintah yang digunakan bersifat *case-insensitive*. Sebaiknya anda membiasakan menggunakan huruf kapital untuk perintah SQL. Kebiasaan ini untuk membedakan nama database, nama tabel, atau nama kolom (Eko Priyo Utomo: 2014;58-59).

MySQL merupakan sistem basis data relasional dimana item data diorganisasikan dalam bentuk tabel. Untuk menciptakan sebuah tabel, sebuah *database* harus dibuat terlebih dahulu (Riyanto, 2014;63).

## II.8. Rancangan Database

*Database* dapat didefinisikan sebagai kumpulan dari item data (*file* atau tabel) yang saling berhubungan satu dengan yang lainnya yang diorganisasikan berdasarkan sebuah skema atau struktur tertentu, tersimpan dalam perangkat keras komputer, dan digunakan per perangkat lunak untuk memanipulasinya. Sedangkan, kenapa *database* diperlukan, setidaknya alasan berikut dapat menjawabnya.

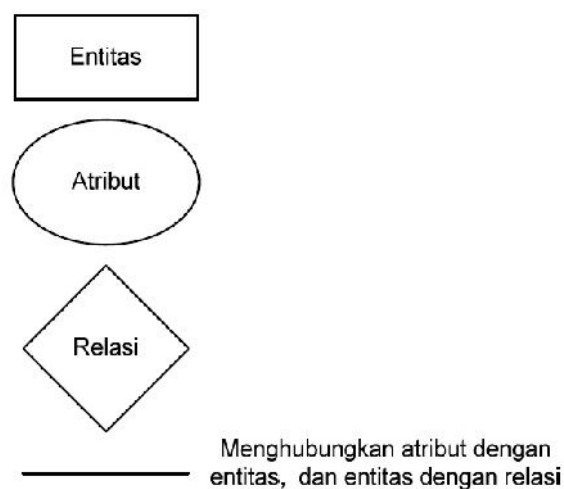
1. *Database* merupakan salah satu komponen penting dalam sistem informasi, karena merupakan dasar dalam menyediakan informasi.
2. Mampu menentukan kualitas informasi, meliputi: akurasi, tepat pada waktunya, dan relevan. Karena informasi dapat dikatakan bernilai apabila manfaatnya lebih besar dibanding dengan biaya yang dikeluarkan.
3. Dapat mereduksi duplikasi data (*data redundancy*) dan mengurangi pemborosan tempat simpanan luar.

Proses perancangan *database* merupakan tahap penting agar sistem yang dikembangkan efisien dalam penggunaan ruang penyimpanan, pengaksesan data, terjaminnya integritas data, dan mudah dalam pemanipulasiannya. Beberapa metode populer yang digunakan untuk melakukan pemodelan data adalah Diagram ER (*Entity Relationship*) dan Normalisasi (Riyanto: 2014;61).

### II.8.1. Entity Relationship Diagram (ERD)

*Entity relationship* (ER) data model didasarkan pada persepsi terhadap dunia nyata yang tersusun atas kumpulan objek-objek dasar yang disebut entitas dan hubungan antar objek (Simarmata & Paryudi: 2006:59). Entitas adalah

sesuatu atau objek dalam dunia nyata yang dapat dibedakan dari objek lain. Misal: mahasiswa, dan matakuliah. Entitas digambarkan dalam basis data dengan kumpulan atribut. Misalnya: nim, nama, alamat, dan kota. Relasi adalah hubungan antara beberapa entitas. Misalnya: relasi menghubungkan mahasiswa dengan mata kuliah yang diambilnya. Struktur logis (skema *database*) dapat ditunjukkan secara grafis dengan diagram ER yang dibentuk dari komponen-komponen berikut :



**Gambar II.4. Komponen ERD**  
(Sumber : D. Tri Octafian ; 2011)

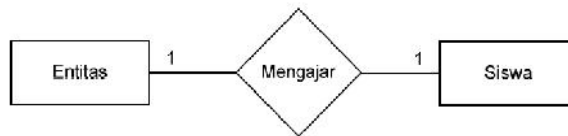
## 1. Pemetaan kardinalitas

Pemetaan kardinalitas menyatakan jumlah entitas di mana entitas lain dapat dihubungkan ke entitas tersebut melalui sebuah himpunan relasi.

### a. *One to One*

Sebuah entitas pada A berhubungan dengan paling banyak satu entitas pada B dan sebuah entitas pada B berhubungan dengan paling banyak satu entitas pada A. Contohnya pada pengajaran privat, satu guru satu

siswa. Seorang guru mengajar seorang siswa, seorang siswa diajar oleh seorang guru.



**Gambar II.5. Hubungan *One to One***  
(Sumber : D. Tri Octafian ; 2011)

**b. *One to Many/ Many to One***

Sebuah entitas pada A berhubungan dengan lebih dari satu entitas pada B dan sebuah entitas pada B berhubungan dengan paling banyak satu entitas pada A, atau sebaliknya (*Many to One*). Contohnya dalam satu perusahaan, satu bagian mempekerjakan banyak pegawai. Satu bagian mempekerjakan banyak pegawai, satu pegawai kerja dalam satu bagian.

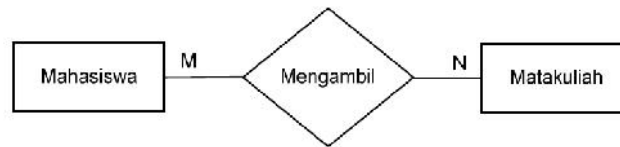


**Gambar II.6. Hubungan *One to Many***  
(Sumber : D. Tri Octafian ; 2011)

**c. *Many To Many***

Sebuah entitas pada A berhubungan dengan lebih dari satu entitas pada B dan sebuah entitas pada B berhubungan dengan lebih dari satu entitas pada A. Contohnya dalam universitas, seorang mahasiswa dapat mengambil banyak mata kuliah. Satu mahasiswa mengambil banyak

mata kuliah dan satu mata kuliah diambil banyak mahasiswa (D. tri Octafian, 2011 ; 150-152).



**Gambar II.7. Hubungan *Many to Many***  
(Sumber : D. Tri Octafian ; 2011)

## II.8.2. Normalisasi Database

Sebuah rancangan database dapat dikategorikan kurang baik, apabila:

1. Data disimpan di beberapa tempat atau *record*.
2. Adanya redundansi atau pengulangan data sehingga memboroskan ruang penyimpanan serta menyulitkan ketika melakukan *update* atau perubahan data.
3. Timbul nilai null (*null value*).
4. Menimbulkan informasi yang tidak bermanfaat.

Oleh karena itu, normalisasi perlu dilakukan. Normalisasi database membuat struktur database yang dihasilkan semakin optimal. Intinya, proses ini menghindarkan pengguna dari redundansi data. Normalisasi database biasanya dilakukan pada database yang berskala besar (Eko Priyo Utomo: 2014;72-73).

### 1. Normalisasi 1 (1NF)

Dalam mencegah berbagai macam anomali, dapat menormalkan tabel menjadi dua tabel bentuk normalisasi 1 (1NF). Aturan yang ada dalam 1NF adalah sebuah baris data tidak dapat terdiri dari kumpulan data yang sama, masing-masing kolom harus mempunyai nilai yang unik, dan masing-masing baris mempunyai *identifier* unik yaitu *primary key*.

**Tabel II.1. Data Normalisasi Bentuk 1 (1NF)**

Pelanggan_Id	Pelanggan_Nama	Matkul_Id	Matkul_Mhs_Id	Barang_Detail
201	Amir	50	201	Rekayasa perangkat lunak
202	Agus	51	201	Komunikasi data
203	Bobby	52	202	Manajemen proyek
204	Cahya	53	203	Rekayasa perangkat lunak
205	Joko	54	204	Penambangan data
		55	205	Manajemen strategis

(Sumber : Eko Priyo Utomo ; 2014)

Penggabungan antara kolom *Matkul\_Id* dan *Matkul\_Mhs\_Id* merupakan *primary key* (Eko Priyo Utomo: 2014;74-75).

### 2. Normalisasi 2 (2NF)

Normalisasi bentuk kedua (2NF) harus memenuhi syarat 1NF dan tidak ada ketergantungan *primary key*.

**Tabel II.2. Data Pelanggan**

Pelanggan_Id	Pelanggan_Nama	Order_Id	Order_Nama	Detail
201	Amir	60	Jeruk	Jeruk Bali
202	Amir	61	Apel	Apel Malang
203	Bobby	62	Bakpia	Bakpia Yogya
204	Cahya	63	Tahu	Tahu Sumedang

(Sumber : Eko Priyo Utomo ; 2014)

Pada tabel pelanggan diatas, penggabungan antara Pelanggan\_Id dan Ordeer\_Id merupakan *primary key*. Tabel pelanggan diatas memenuhi persyaratan normalisasi pertama (1NF) namun belum 2NF karena ada ketergantungan parsial kolom pada *primary key*. Kolom Pelanggan\_Nama hanya bergantung pada Pelanggan\_Id, Order\_Nama bergantung pada Order\_Id dan tidak ada link/hubungan antara Detail dan Pelanggan\_Nama (Eko Priyo Utomo: 2014;75).

**Tabel II.3. Data Normalisasi Bentuk 2 (2NF)**

Pelanggan_Id	Pelanggan_Nama	Order_Id	Order_Nama
201	Amir	60	Jeruk Bali
202	Bobby	61	Apel Malang
203	Cahya	62	Bakpia Yogya
		63	Tahu Sumedang

Pelanggan_Id	Order_Nama	Barang_Detail
201	60	Jeruk
201	61	Apel
202	62	Bakpia
203	63	Tahu

(Sumber : Eko Priyo Utomo ; 2014)

### 3. Normalisasi 3 (3NF)

Normalisasi bentuk ketiga (3NF) dapat dibuat dengan kolom-kolom yang bukan primer tergantung pada *primary key*. Ketergantungan fungsional transitif seharusnya dikeluarkan dari tabel, tabel memenuhi persyaratan normalisasi kedua (2NF) (Eko Priyo Utomo: 2014;76).

**Tabel II.4. Data Normalisasi Bentuk 3 (3NF)**

Mahasiswa_Id	Mahasiswa_Nama	Mahasiswa_KodePos
10	Melani	44123
11	Nania	64111
12	Rosnia	22780

Mahasiswa_KodePos	Mahasiswa_Jalan	Mahasiswa_Kota	Mahasiswa_Kabupaten
44123	Jl. Kucing 12	Condong Catur	Sleman
64111	Jl. Harimau 21	Keras	Kediri
22780	Jl. Pelanduk 33	Nagrek	Sumedang

(Sumber : Eko Priyo Utomo ; 2014)

## II.9. Unified Modelling Language (UML)

*Unified Modelling Language* (UML) adalah suatu alat untuk memvisualisasikan dan mendokumentasikan hasil analisa dan desain yang berisi sintak dalam memodelkan sistem secara visual (Braun, *et. al.* 2001). Juga merupakan satu kumpulan konvensi pemodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem *software* yang terkait dengan objek (Whitten, *et. al.* 2004).

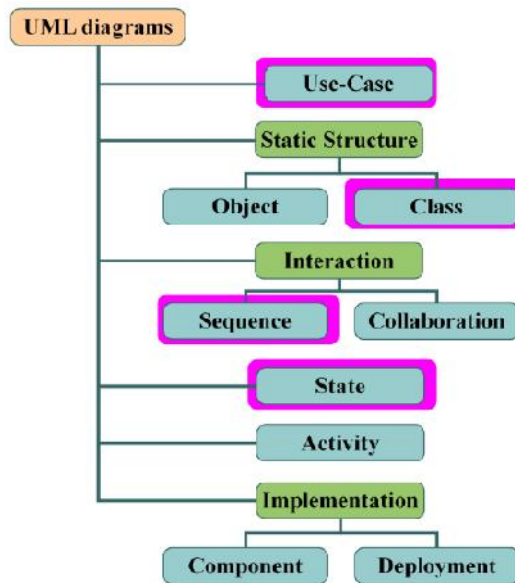
Sejarah UML sendiri terbagi dalam dua fase; sebelum dan sesudah munculnya UML. Dalam fase sebelum, UML sebenarnya sudah mulai

diperkenalkan sejak tahun 1990an namun notasi yang dikembangkan oleh para ahli analisis dan desain berbeda-beda, sehingga dapat dikatakan belum memiliki standarisasi.

Fase kedua; dilandasi dengan pemikiran untuk mempersatukan metode tersebut dan dimotori oleh Object Management Group (OMG) maka pengembangan UML dimulai pada akhir tahun 1994 ketika Grady Booch dengan metode OOD (*Object-Oriented Design*), Jim Rumbaugh dengan metode OMT (*Object Modelling Technique*) mereka ini bekerja pada Rational Software Corporation dan Ivar Jacobson dengan metode OOSE (*Object-Oriented Software Engineering*) yang bekerja pada perusahaan Objectory Rational.

Saat ini sebagian besar para perancang sistem informasi dalam menggambarkan informasi dengan memanfaatkan UML diagram dengan tujuan utama untuk membantu tim proyek berkomunikasi, mengeksplorasi potensi desain, dan memvalidasi desain arsitektur perangkat lunak atau pembuat program. Secara filosofi UML diilhami oleh konsep yang telah ada yaitu konsep permodelan *Object Oriented* karena konsep ini menganalogikan sistem seperti kehidupan nyata yang didominasi oleh obyek dan digambarkan atau dinotasikan dalam simbol-simbol yang cukup spesifik (Haviluddin: 2011;1-2).

Berikut gambar dari diagram UML :



**Gambar II.8. Diagram UML**  
(Sumber : Haviluddin, 2011)

Tujuan dari penggunaan diagram UML seperti diungkapkan oleh Schmuller J. (2004), *“The purpose of the diagrams is to present multiple views of a system; this set of multiple views is called a model”*. Berikut tujuan utama dalam desain UML adalah (Sugrue J. 2009) :

1. Menyediakan bagi pengguna (analisis dan desain sistem) suatu bahasa pemodelan visual yang ekspresif sehingga mereka dapat mengembangkan dan melakukan pertukaran model data yang bermakna.
2. Menyediakan mekanisme yang spesialisasi untuk memperluas konsep inti.
3. Karena merupakan bahasa pemodelan visual dalam proses pembangunannya maka UML bersifat independen terhadap bahasa pemrograman tertentu.
4. Memberikan dasar formal untuk pemahaman bahasa pemodelan.

5. Mendorong pertumbuhan pasar terhadap penggunaan alat desain sistem yang berorientasi objek (OO).
6. Mendukung konsep pembangunan tingkat yang lebih tinggi seperti kolaborasi, kerangka, pola dan komponen terhadap suatu sistem.
7. Memiliki integrasi praktik terbaik.

### II.9.1. Use Case Diagram

*Use case* adalah rangkaian atau uraian sekelompok yang saling terkait dan membentuk sistem secara teratur yang dilakukan atau diawasi oleh sebuah aktor. *Use case* digunakan untuk membentuk tingkah laku benda dalam sebuah model serta direalisasikan oleh sebuah kolaborasi.

Hal yang ditekankan pada diagram ini adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem. *Use case* menyatakan sebuah aktivitas atas pekerjaan tertentu. Aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu.

Sebuah *use case* dapat di-*include* oleh lebih dari satu *use case* lain. Oleh karena itu, duplikasi fungsionalitas dapat dihindari dengan cara menarik keluar fungsionalitas yang *common*. Sebuah *use case* juga dapat meng-*extend use case* lain dengan *behavior*-nya sendiri. Hubungan generalisasi antar *use case* menunjukkan bahwa *use case* merupakan spesialisasi dari yang lain (Hamim Tohari: 2014;47-48).

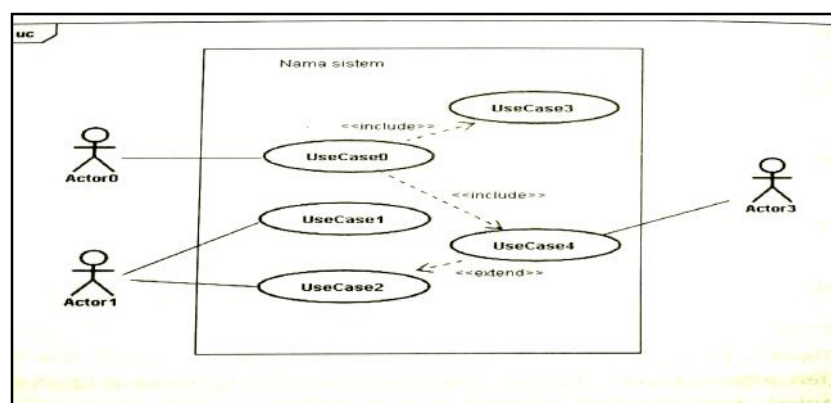
## 1. Menyusun Diagram Use Case

Langkah-langkah yang dibutuhkan untuk menyusun diagram *use case* :

- Mengidentifikasi pelaku bisnis.
- Mengidentifikasi *use case* persyaratan bisnis.
- Membuat diagram model *use case*.
- Mendokumentasikan naratif *use case* persyaratan bisnis (Hamim Tohari: 2014;49).

## 2. Elemen-Elemen Diagram Use Case

Beberapa elemen yang digunakan pada diagram *use case* dapat dilihat pada gambar II.9 berikut.



**Gambar II.9. Elemen-Elemen Diagram Use Case**  
(Sumber : Hamim Tohari ; 2014)

- Sistem, menyatakan batasan sistem dalam relasi dengan *actor-actor* yang menggunakannya (di luar sistem) dan fitur-fitur yang harus disediakan (dalam sistem). Sistem digambarkan dengan segi empat yang membatasi semua *use case* dalam sistem terhadap pihak mana sistem akan berinteraksi.

- b. *Actor* atau actor, dapat berupa manusia, sistem, atau *device* yang memiliki peranan dalam keberhasilan operasi dari sistem. Digambarkan dengan *icon* yang mungkin bervariasi namun konsepnya sama.
- c. *Use case*, mengidentifikasi fitur kunci dari sistem. Tanpa fitur ini, sistem tidak akan memenuhi permintaan *user/actor*. Setiap *use case* mengekspresikan *goal* dari sistem yang harus dicapai. Diberi nama sesuai dengan *goal*-nya dan digambarkan dengan elips (dengan nama didalamnya).
- d. *Association*, mengidentifikasikan interaksi antara setiap *actor* tertentu dengan setiap *use case* tertentu. Digambarkan dengan garis antara *actor* terhadap *use case* yang bersangkutan. Asosiasi bias berarah (garis dengan anak panah) jika komunikasi satu arah, namun umumnya terjadi kedua arah (tanpa anak panah) karena selalu diperlukan demikian.
- e. *Stereotype*, memungkinkan perluasan UML tanpa memodifikasinya. Berperan sebagai kualifier pada suatu elemen modal. Menyediakan informasi lebih banyak mengenai peranan dari elemen tanpa menyebutkan implementasinya.
- f. *Dependency*, dependensi <<include>> mengidentifikasi hubungan antar dua *use case* dimana yang satu memanggil yang lain, digambarkan dengan garis putus-putus bermata panah dengan notasi <<include>> pada garis, dan arah mata panah sesuai dengan arah pemanggilan. Dependensi <<extend>> jika pemanggilan memerlukan adanya kondisi tertentu maka berlaku dependensi <<extend>> dan digambarkan serupa

dengan dependensi <<include>> kecuali arah panah berlawanan (Hamim Tohari: 2014;51-54).

### II.9.2. Class Diagram

Kelas (*Class*) adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan perancangan berorientasi objek. Kelas menggambarkan keadaan (atribut/property) suatu system, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metode/fungsi). Dalam pemodelan statis dari sebuah sistem, diagram kelas biasanya digunakan untuk memodelkan salah satu dari tiga hal berikut :

1. Perbendaharaan dari sistem.
2. Kolaborasi.
3. Skema basis data *logical*.

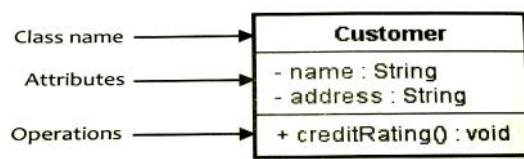
Kelas memiliki tiga area pokok :

1. Nama (dan *stereotype*)
2. Atribut
3. Metode atau operasi

Atribut dan metode dapat memiliki salah satu sifat berikut :

1. *Private*, tidak dapat dipanggil dari luar *class* yang bersangkutan.
2. *Protected*, hanya dapat dipanggil oleh *class* yang bersangkutan dan anak-anak yang mewarisinya.
3. *Public*, dapat dipanggil oleh siapa saja.

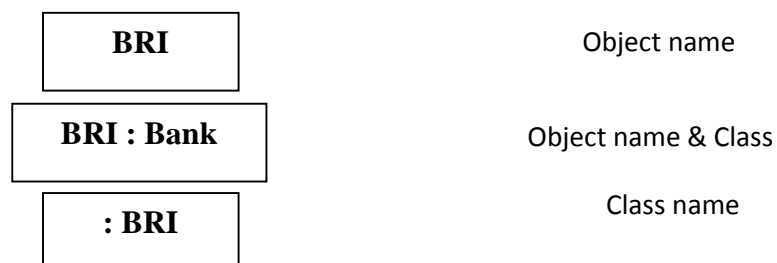
Kelas dapat berupa implementasi dari sebuah *interface*, yaitu *class* abstrak yang hanya memiliki metode. *Interface* tidak dapat langsung diinstansiasikan, tetapi harus diimplementasikan dahulu menjadi sebuah *class*. Dengan demikian *interface* mendukung resolusi metode pada saat *run-time* (Hamim Tohari: 2014;83-84).



**Gambar II.10. Contoh Kelas**  
(Sumber : Hamim Tohari ; 2014)

### II.9.3. Sequence Diagram

*Sequence Diagram* menggambarkan interaksi antara sejumlah objek dalam urutan waktu. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara objek juga interaksi antar objek yang terjadi pada titik tertentu dalam eksekusi system. Dalam UML, objek pada diagram *sequence* digambarkan dengan segi empat, yang berisi nama dari objek yang digarisbawahi. Terdapat 3 cara untuk menamai objek yaitu, nama objek, nama objek dan *class* serta nama *class*. Misalnya :



**Gambar II.11. Penamaan Objek pada Sequence Diagram**  
(Sumber : Hamim Tohari ; 2014)

Pada *diagram sequence*, setiap objek hanya memiliki garis yang digambarkan garis putus-putus ke bawah. Pesan antar objek digambarkan dengan anak panah dari objek yang mengirimkan pesan ke objek yang menerima pesan (Hamim Tohari: 2014;101).

#### **II.9.4. Activity Diagram**

*Activity diagram* memodelkan *workflow* proses bisnis dan urutan aktivitas dalam sebuah proses. Diagram ini sangat mirip dengan *flowchart* karena memodelkan *workflow* dari satu aktivitas ke aktivitas lainnya atau dari aktivitas ke status. Membuat *activity diagram* pada awal pemodelan proses cukup menguntungkan untuk membantu memahami keseluruhan proses. *Activity diagram* juga bermanfaat untuk menggambarkan *parallel behaviour* atau menggambarkan interaksi antara beberapa *use case*.

##### **1. Elemen-Elemen Activity Diagram**

- a. Status *Start* (mulai) dan *end* (akhir).
- b. Aktivitas yang mempresentasikan sebuah langkah dalam *workflow*.
- c. *Transition* menunjukkan terjadinya perubahan status aktivitas (*transition show what State follows another*).
- d. Keputusan yang menunjukkan *alternative* dalam *workflow*.
- e. *Synchronization bars* yang menunjukkan *subflow parallel*. *Synchronization bars* dapat digunakan untuk menunjukkan *concurrent threads* pada *workflow* proses bisnis.

- f. *Swimlanes* yang mempresentasikan *role* bisnis yang bertanggung jawab pada aktivitas yang berjalan (Hamim Tohari: 2014;114-115).