

BAB II

TINJAUAN PUSTAKA

II.1. Konsep Sistem Informasi

Konsep dasar sistem informasi akan menguraikan beberapa pengertian sistem, karakteristik sistem, pengertian dan komponen sistem informasi.

II.1.1. Sistem

Menurut Kusriani & Andi Koniyo (2007 : 5), kata sistem mempunyai beberapa pengertian, tergantung dari sudut pandang mana kata tersebut didefinisikan. Secara garis besar ada dua kelompok pendekatan, yaitu :

1. Pendekatan sistem yang lebih menekankan pada elemen-elemen atau kelompoknya, yang dalam hal ini sistem itu didefinisikan sebagai suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu aturan tertentu.
2. Pendekatan sistem sebagai jaringan kerja dari prosedur, yang lebih menekankan urutan operasi di dalam sistem. Prosedur (*procedure*) didefinisikan oleh F. Nueshl sebagai urutan operasi kerja (tuliskan-menulis), yang biasanya melibatkan beberapa orang di dalam satu atau lebih departemen, yang diterapkan untuk menjamin penanganan yang seragam dari transaksi bisnis yang terjadi.

Pendekatan sistem yang lebih menekankan pada elemen-elemen atau komponennya mendefinisikan sistem sebagai sekumpulan elemen yang saling terkait atau terpadu yang dimaksudkan untuk mencapai suatu tujuan. Dengan demikian di dalam suatu sistem, komponen-komponen ini tidak dapat berdiri sendiri-sendiri, tetapi sebaliknya, saling berhubungan hingga membentuk satu kesatuan sehingga tujuan sistem itu dapat tercapai.

II.1.2. Karakteristik Sistem

Menurut Kusriani & Andi Koniyo (2007 : 6), sistem mempunyai beberapa karakteristik atas sifat-sifat tertentu, antara lain :

1. **Komponen Sistem (*Component*)**

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang saling bekerja sama membentuk suatu komponen sistem atau bagian-bagian dari sistem.

2. **Batasan Sistem (*Boundary*)**

Merupakan daerah yang membatasi suatu sistem dengan sistem yang lain atau dengan lingkungan kerjanya.

3. **Subsistem**

Bagian-bagian dari sistem yang beraktivitas dan berinteraksi satu sama lain untuk mencapai tujuan dengan sasarannya masing-masing.

4. **Lingkungan Luar Sistem (*Environment*)**

Suatu sistem yang ada di luar dari batas sistem yang dipengaruhi oleh operasi sistem.

5. Penghubung Sistem (*Interface*)

Media penghubung antara suatu subsistem dengan subsistem lain. Adanya penghubung ini memungkinkan berbagai sumber daya mengalir dari suatu subsistem ke subsistem lainnya.

6. Masukan Sistem (*Input*)

Energi yang masuk ke dalam sistem, berupa perawatan dan sinyal.

Masukan perawatan adalah energi yang dimasukkan supaya sistem tersebut dapat berinteraksi.

7. Keluaran Sistem (*Output*)

Hasil energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan.

8. Pengolahan Sistem (*Process*)

Suatu sistem dapat mempunyai suatu bagian pengolahan yang akan mengubah masukan menjadi keluaran.

9. Sasaran Sistem (*Object*)

Tujuan yang ingin dicapai oleh sistem, akan dikatakan berhasil apabila mengenai sasaran atau tujuan.

II.1.3. Klasifikasi Sistem

Menurut Kusri & Andi Koniyo (2007 : 7), suatu sistem dapat diklasifikasikan menjadi seperti berikut :

1. Sistem abstrak dan fisik

Sistem abstrak adalah suatu sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik, sedangkan sistem fisik adalah sistem yang ada secara fisik

2. Sistem alamiah dan sistem buatan manusia

Sistem alamiah adalah sistem yang terjadi melalui proses alam sedangkan sistem buatan manusia adalah sistem yang dirancang oleh manusia.

3. Sistem tertentu dan sistem tak tentu

Sistem tertentu adalah suatu sistem yang operasinya dapat diprediksi secara tepat sedangkan sistem tak tertentu adalah sistem dengan perilaku ke depan yang tidak dapat diprediksi.

4. Sistem tertutup dan sistem terbuka

Sistem tertutup adalah sistem yang tidak terpengaruh oleh lingkungan luar atau otomatis, sedangkan sistem terbuka adalah sistem yang berhubungan dan berhubungan oleh lingkungan luar.

II.1.4. Pengertian Informasi

Menurut Kusriani & Andi Koniyo (2007 : 7), informasi adalah data yang sudah diolah menjadi sebuah bentuk yang berarti bagi pengguna, yang bermanfaat dalam pengambilan keputusan saat ini atau mendukung sumber informasi. Data belum memiliki nilai, sedangkan informasi sudah memiliki nilai. Informasi dikatakan bernilai bila manfaatnya lebih besar dibandingkan biaya untuk mendapatkannya.

II.1.5. Kualitas Informasi

Menurut Kusrini & Andi Koniyo (2007 : 8), informasi yang berkualitas memiliki 3 karakteristik, yaitu :

1. Akurat (*Accurate*)

Informasi harus bebas dari kesalahan, tidak bias ataupun menyesatkan. Akurat juga berarti bahwa informasi itu harus dapat dengan jelas mencerminkan maksudnya.

2. Tepat pada waktunya (*timeliness*)

Informasi yang datang pada penerima tidak boleh terlambat, didalam pengambilan keputusan, informasi yang sudah usang tidak lagi bernilai. Bila informasi datang terlambat sehingga pengambilan keputusan terlambat dilakukan, hal itu dapat berakibat fatal bagi perusahaan.

3. Relevan (*relevance*)

Informasi yang disampingkan harus mempunyai keterkaitan dengan masalah yang akan dibahas dengan informasi tersebut. Informasi harus bermanfaat bagi pemakai, disamping karakteristik, nilai informasi juga ikut menentukan kualitasnya. Nilai informasi (*value of information*) ditentukan oleh dua hal, yaitu manfaat dan biaya untuk mendapatkannya. Suatu informasi dikatakan bernilai bila manfaatnya lebih besar dibandingkan biaya untuk mendapatkannya.

II.1.6. Sistem Informasi

Menurut Kusrini & Andi Koniyo (2007 : 8), untuk menghasilkan informasi yang berkualitas maka dibuatkanlah sistem informasi. Sistem informasi didefinisikan oleh Robert A. Litch dan K. Roscoe Bavis sebagai berikut :
”Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan.”

Defenisi umum sistem informasi adalah : ”Sebuah sistem yang terdiri atas rangkaian subsistem informasi terhadap pengolahan data untuk menghasilkan informasi yang berguna dalam pengambilan keputusan.”

II.1.7. Komponen Sistem Informasi

Menurut Kusrini & Andi Koniyo (2007 : 9), dalam suatu sistem informasi terdapat komponen-komponen sebagai berikut :

1. Perangkat keras (*hardware*), mencakup berbagai peranti fisik seperti komputer dan printer.
2. Perangkat lunak (*software*) atau program, yaitu sekumpulan instruksi yang memungkinkan perangkat keras memproses data.
3. Prosedur, yaitu sekumpulan aturan yang dipakai untuk mewujudkan pemrosesan data dan pembangkitan keluaran yang dikehendaki.
4. Orang, yaitu semua pihak yang bertanggung jawab dalam pengembangan sistem informasi, pemrosesan dan penggunaan keluaran sistem informasi.

5. Basis data (*database*), yaitu sekumpulan tabel, hubungan dan lain-lain yang berkaitan dengan penyimpanan data.
6. Jaringan komputer dan komunikasi data, yaitu sistem penghubung yang memungkinkan sumber (*resources*) dipakai secara bersama atau diakses oleh sejumlah pemakai.

II.1.8. Sistem Informasi Akuntansi

Menurut Anastasia Diana & Lilis Setiawati (2011 : 4), Sistem Informasi Akuntansi adalah sistem yang bertujuan untuk mengumpulkan dan memproses data serta melaporkan informasi yang berkaitan dengan transaksi-transaksi keuangan. Misalnya, salah satu input dari Sistem Informasi Akuntansi pada sebuah toko baju, adalah transaksi penjualan, memproses transaksi dengan mencatat penjualan tersebut kedalam jurnal penjualan, mengklasifikasikan transaksi dengan menggunakan kode rekening, dan memposting transaksi ke dalam jurnal. Kemudian secara periodik Sistem Informasi Akuntansi akan menghasilkan output berupa laporan keuangan yang terdiri dari Neraca dan Laporan Laba Rugi.

Menurut Kusriani & Andi Koniyo (2007 : 10), Sistem Informasi akuntansi merupakan sebuah sistem informasi yang mengubah data transaksi bisnis menjadi informasi keuangan yang berguna bagi pemakainya.

Menurut Anastasia Diana & Lilis Setiawati (2011 : 5), Manfaat atau Tujuan dari Sistem Informasi Akuntansi adalah sebagai berikut :

1. Mengamankan harta / kekayaan perusahaan. Meliputi kas perusahaan, persediaan barang dagangan, termasuk aset tetap perusahaan.
2. Mendukung beragam informasi untuk pengambilan. Misalnya, pengelola toko swalayan memerlukan informasi mengenai barang yang diminati oleh konsumen.
3. Menghasilkan informasi untuk pihak *external*, setiap pengelola usaha memiliki kewajiban untuk membayar pajak, besarnya pajak yang dibayar tergantung pada omset penjualan (jika pengelola memilih menggunakan norma dalam perhitungan pajaknya) atau tergantung pada laba rugi usaha.
4. Menghasilkan informasi untuk penilaian kinerja karyawan atau divisi. Sistem informasi dapat juga dimanfaatkan untuk penilaian kinerja karyawan atau divisi, sebagai contoh, pengelola toko swalayan dapat memanfaatkan data penjualan untuk menilai kinerja kasir, kasir mana yang lebih cepat dan lebih cermat dalam melayani pelanggan.
5. Menyediakan data masa lalu untuk kepentingan audit (pemeriksaan). Data yang tersimpan dengan baik sangat memudahkan proses audit (pemeriksaan), satu hal yang penting, audit bukan eksklusif milik perusahaan publik.

Menurut Kusrini & Andi Koniyo (2007 : 10), komponen-komponen yang terdapat dalam sistem informasi akuntansi adalah sebagai berikut:

1. Orang-orang yang mengoperasikan sistem tersebut.
2. Prosedur-prosedur, baik manual maupun yang terotomatisasi, yang dilibatkan dalam pengumpulan, pemrosesan dan penyimpanan data aktivitas-aktivitas organisasi.

3. Data tentang proses-proses bisnis.
4. Software yang dipakai untuk memproses data organisasi.
5. Infrastruktur teknologi informasi.

Didalam organisasi, sistem informasi akuntansi berfungsi untuk :

1. Mengumpulkan dan menyimpan aktivitas yang dilaksanakan disuatu organisasi, sumber daya yang dipengaruhi oleh aktivitas-aktivitas tersebut dan para pelaku aktivitas tersebut.
2. Mengubah data menjadi informasi yang berguna bagi manajemen.
3. Menyediakan pengendalian yang memadai.

Menurut Kusri & Andi Koniyo (2007 : 10), sistem informasi akuntansi merupakan pendukung aktivitas organisasi, yang termasuk pendukung aktivitas organisasi adalah:

1. Infrastruktur perusahaan : akuntansi, hukum, administrasi umum.
2. Sumber daya manusia : perekrutan, pengontrolan, pelatihan dan kompensasi kepada pegawai.
3. Teknologi : peningkatan produk dan jasa (penelitian).
4. Pembelian.

Sementara itu aktivitas utamanya adalah:

1. *Inbound Logistics* : Penerimaan, penyimpanan dan distribusi bahan-bahan masukan.
2. Operasi : aktivitas untuk mengubah masukan menjadi barang atau jasa.
3. *Outbound Logistics* : distribusi produk ke pelanggan.
4. Pemasaran dan Penjualan.

5. Pelayanan : Dukungan purna jual dan *maintenance*.

Dari penjelasan tersebut dapat diambil kesimpulan bahwa akuntansi adalah proses pengidentifikasian, pengukuran dan melaporkan informasi ekonomi untuk memungkinkan adanya pembuatan pertimbangan dan keputusan bagi yang menggunakan informasi.

II.2. Pengertian Retur Penjualan

Menurut Partiwi Dwi Astuti (2012 : 206), retur penjualan merupakan pengembalian barang dagang yang telah dibeli kepada penjual karena barang dagang yang dikirim tidak sesuai dengan spesifikasi yang ditentukan oleh pembeli. Penjual dapat juga memberikan sejumlah potongan atas barang yang dijualnya karena barang cacat, rusak atau sebab lainnya. Dengan demikian, maka potongan penjualan merupakan penyesuaian harga yang diberikan penjual akibat barang dagang yang dikirim ke pembeli tidak sesuai dengan spesifikasi yang dipesannya.

II.2.1. Retur Penjualan dan Pengurangan Harga

Menurut Ais Zakiyudin (2013 : 47), apabila dalam penjualan barang dagang pihak penjual setuju menerima kembali barang yang dijualnya atau setuju mengurangi dari harga jualnya atas barang yang dikirimkan tidak sesuai dengan keinginan pembeli, maka dicatat dalam perkiraan yang disebut **Retur Penjualan dan Pengurangan Harga** (*Sales Return and Allowance*).

Untuk mencatat transaksi penerimaan kembalian barang dagang dan pemberian pengurangan harga adalah debit perkiraan Retur Penjualan dan Pengurangan Harga, dan kredit perkiraan Piutang Dagang.

Retur Penjualan dan Pengurangan Harga (<i>Sales Return and Allowance</i>)	Rp. xxx	
Piutang Dagang (<i>Acc. Receivable</i>)		Rp. xxx

Gambar II.1. Jurnal Umum Retur Penjualan dan Pengurangan Harga
(Sumber : Ais Zakiyudin, 2013 : 47)

Contoh Transaksi :

Tanggal 27 Juni 2010 dikirim nota kredit kepada Toko Ria, Solo atas penerimaan kembalian barang dagangan seharga Rp. 200.000,00 karena kualitasnya berbeda.

Transaksi diatas akan dicatat dalam Jurnal Umum, akan terlihat sebagai berikut :

Tabel II.1. Jurnal Umum Retur Penjualan dan Pengurangan Harga

Tanggal	Keterangan	Ref	Debit	Kredit
27 Juni	Retur Penjualan dan Pengurangan Harga (<i>Sales Return and Allowance</i>) Piutang Dagang (<i>Acc. Receivable</i>)		200.000,00	200.000,00

(Sumber : Ais Zakiyudin, 2013 : 47)

II.3. Basisdata (*Database*)

Basisdata menurut (Janner Simarmata,dkk ; 2006 : 1) yang dikutip dari pendapat Stephens dan Plew (2000), adalah mekanisme yang digunakan untuk menyimpan informasi atau data. Informasi adalah sesuatu yang kita gunakan

sehari-hari untuk berbagai alasan. Dengan basisdata, pengguna dapat menyimpan data secara terorganisasi. Setelah data disimpan, informasi harus mudah diambil. Kriteria dapat digunakan untuk mengambil informasi. Cara data disimpan dalam basisdata menentukan seberapa mudah mencari informasi berdasarkan banyak kriteria. Data pun harus mudah ditambahkan ke dalam basisdata, dimodifikasi dan dihapus.

Menurut (Janner Simarmata,dkk ; 2006 : 1) yang dikutip dari pendapat Silberschatz, dkk (2002) mendefinisikan basisdata sebagai kumpulan data berisi informasi yang sesuai untuk sebuah perusahaan. Sistem manajemen basisdata (DBMS) adalah kumpulan data yang saling berhubungan dan kumpulan program untuk mengakses data. Tujuan utama sistem manajemen basisdata adalah menyediakan cara menyimpan dan mengambil informasi basisdata secara mudah dan efisien (Janner Simarmata,dkk ; 2006 : 1).

II.3.1. Komponen Penyusunan Basisdata

Komponen penyusunan basisdata adalah :

1. Skema basisdata

Pada dasarnya, skema adalah sekelompok objek dalam basisdata yang saling berhubungan. Dengan skema, objek yang dihubungkan memiliki relasi satu sama lain. Ada satu pemilik skema yaitu yang memiliki akses manipulasi struktur semua objek dalam skema.

2. Tabel

Tabel adalah unit penyimpanan fisik utama untuk data dalam basisdata. Ketika mengakses basisdata, pengguna mengacu tabel untuk data yang diinginkan. Basisdata bisa terdiri atas banyak tabel sehingga harus terdapat relasi antar tabel. Ada 4 macam tabel yaitu :

- a. Tabel data berfungsi menyimpan sebagian besar data yang ditemukan pada basisdata.
- b. Join tabel adalah tabel yang digunakan untuk membentuk relasi antar dua tabel.
- c. Subset tabel mengandung bagian data suatu tabel data.
- d. Tabel validasi, kadang disebut juga tabel kode, digunakan untuk membuat valid data yang dimasukkan pada tabel lainnya.

3. Kolom

Kolom atau field adalah kategori informasi yang terdapat dalam tabel. Kolom bagi tabel seperti atribut bagi entitas. Kolom adalah struktur penyimpanan logis dalam sebuah basisdata. Masing-masing kolom memiliki tipe data.

4. Baris

Suatu baris adalah kumpulan semua kolom yang berhubungan dengan kejadian tunggal. Dengan kata lain, satu baris data adalah record dalam tabel.

5. Tipe Data

Tipe data menentukan tipe data yang disimpan dalam kolom tabel (Janner Simarmata,dkk ; 2010 : 36).

II.3.2. Tujuan Basisdata

Basisdata bertujuan untuk mengatur data sehingga diperoleh kemudahan, ketepatan, kecepatan dalam pengambilan kembali. Untuk mencapai tujuannya, syarat sebuah basisdata adalah sebagai berikut :

1. Tidak adanya redudansi dan inkosistensi data

Redudansi terjadi jika suatu informasi disimpan di beberapa tempat. Akibat dari redudansi adalah inkosistensi data atau data yang tidak konsisten.

2. Kesulitan Pengaksesan Data

Basisdata memiliki fasilitas untuk melakukan pencarian informasi dengan menggunakan *Query* ataupun dari tool untuk melihat tabelnya.

3. *Multiple User*

Basisdata memungkinkan penggunaan data bersama-sama oleh banyak pengguna pada saat yang bersamaan atau pada saat yang berbeda (Kusrini; 2007 : 2 – 7).

II.3.3. Database SQL Server 2008

SQL server 2008 adalah sebuah terobosan baru dari Microsoft dalam bidang database. SQL server adalah sebuah DBMS (*Database Management System*) yang dibuat oleh Microsoft untuk ikut berkecimpung dalam persaingan dunia pengolahan data menyusun pendahuluannya seperti IBM dan Oracle. SQL server 2008 membawa beberapa terobosan dalam bidang pengolahan dan penyimpanan data (Wahana Komputer ; 2010 : 2).

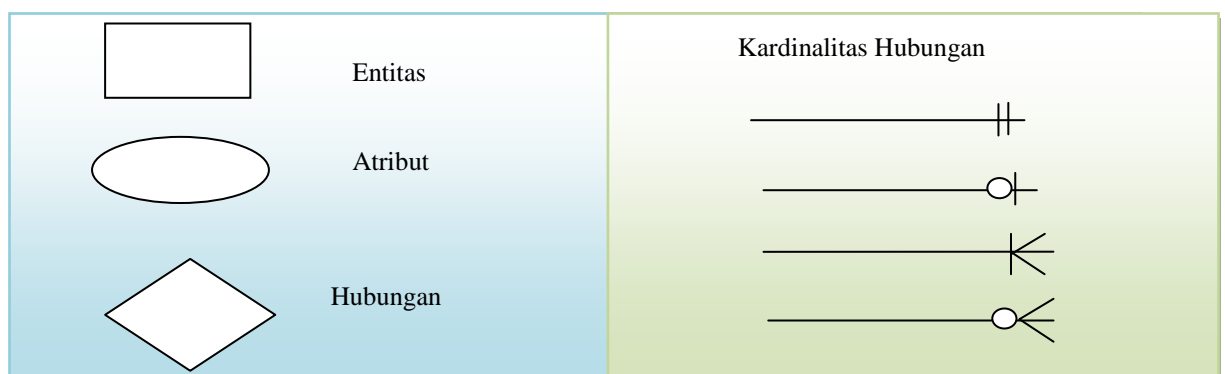
II.4. Entity Relationship Diagram (ERD)

Entity Relationship Diagram adalah alat pemodelan data utama dan akan membantu mengorganisasi data dalam suatu protek kedalam entitas-entitas dan menentukan hubungan antar entitas. Proses memungkinkan analis menghasilkan sturktur basisdata yang baik sehingga data dapat disimpan dan diambil secara efisien (Janner Simarmata,dkk ; 2010 : 67).

II.4.1. Model E-R

Menurut Abdul Kadir (2009:30), model E-R adalah suatu model yang digunakan untuk menggambarkan data dalam bentuk entitas, atribut dan hubungan antar entitas. Huruf E sendiri meyakini entitas dan R menyatakan hubungan (dari kata *Relationship*). Model ini dinyatakan dalam bentuk diagram, itulah sebabnya model E-R sering disebut sebagai diagram E-R.

Model E-R melibatkan sejumlah notasi, beberapa notasi dasar dalam model E-R ditunjukkan pada gambar II.1, notasi-notasi tersebut diberikan hanya untuk memberikan suatu pengetahuan dasar.



Gambar II.2 Sejumlah notasi pada model E-R

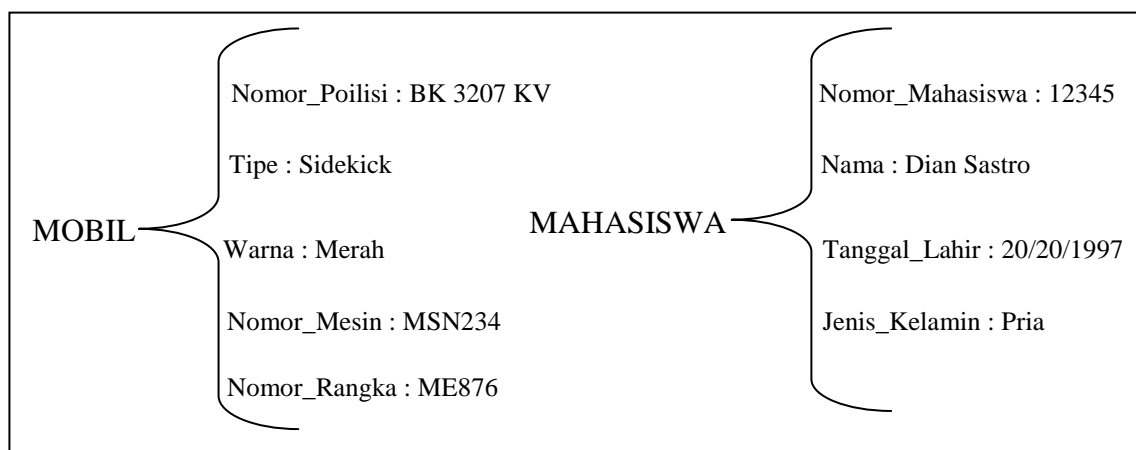
(Sumber : Abdul Kadir; 2009 : 31)

1. Entitas

Yang dimaksud dengan entitas adalah sesuatu dalam dunia nyata yang keberadaannya tidak bergantung pada yang lain. Sebagai contoh, setiap pegawai dalam sebuah organisasi adalah sebuah entitas. Entitas dapat berupa suatu yang nyata ataupun abstrak (berupa suatu konsep). Secara lebih rinci dijelaskan bahwa entitas dapat berupa seseorang, sebuah tempat, sebuah objek, sebuah kejadian atau suatu konsep.

2. Atribut

Setiap entitas dinyatakan dalam sejumlah atribut. Atribut adalah properti atau karakteristik yang terdapat pada setiap entitas. Sebagai contoh, pada gambar II.2, terdapat entitas MOBIL yang mengandung atribut Nomor_Polisi, Tipe, Warna, Nomor_Rangka dan Nomor_Mesin. Selain itu terdapat entitas MAHASISWA yang mengandung atribut Nomor_Mahasiswa, Nama, Tanggal_Lahir, dan Jenis_Kelamin.

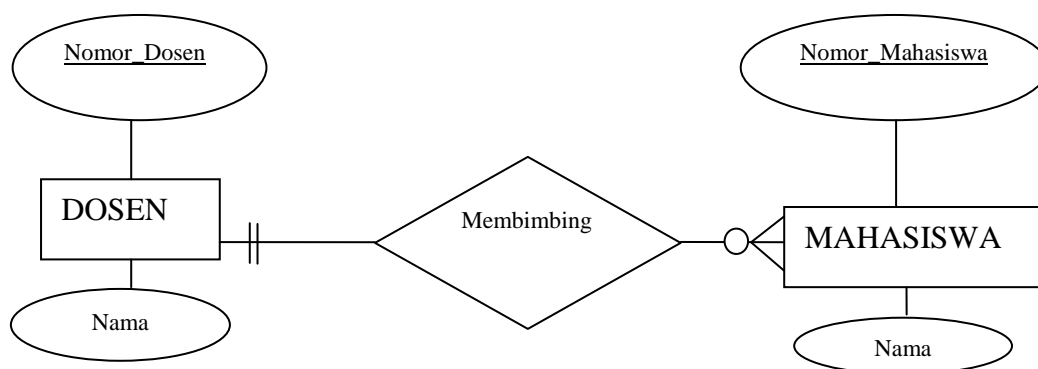


Gambar II.3 Contoh Entitas dan Atribut

(Sumber : Abdul Kadir ; 2009 : 32)

3. Hubungan (*Relationship*)

Hubungan (*Relationship*) menyatakan ketertarikan antara beberapa tipe entitas. Sebagai contoh, tipe entitas MAHASISWA dan DOSEN mempunyai hubungan yang mencerminkan bahwa seorang mahasiswa memiliki dosen pembimbing akademis. Gambar II.3 menunjukkan hubungan tersebut.



Gambar II.4 Contoh Hubungan antara tipe entitas

(Sumber : Abdul Kadir ; 2009 : 45)

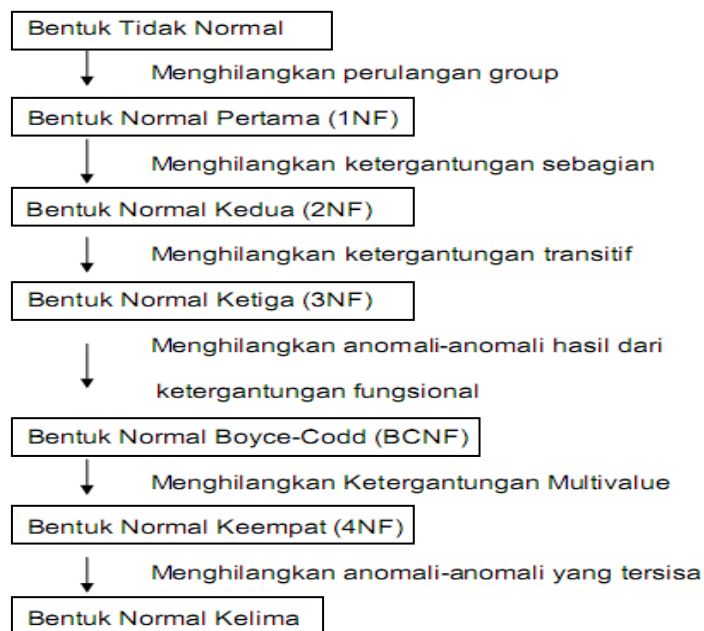
II.5. Normalisasi

Menurut Abdul Kadir (2009:116), normalisasi adalah proses yang digunakan untuk menentukan pengelompokan atribut-atribut dalam sebuah relasi sehingga diperoleh relasi yang berstruktur baik. Dalam hal ini yang dimaksud dengan relasi yang berstruktur baik adalah relasi yang memenuhi dua kondisi berikut

1. Mengandung redundansi sedikit mungkin, dan
2. Memungkinkan baris-baris dalam relasi disisipkan, dimodifikasi dan dihapus tanpa menimbulkan kesalahan atau ketidakkonsistenan.

Normalisasi sendiri dilakukan melalui sejumlah langkah. Setiap langkah berhubungan dengan bentuk normal (*normal form*) tertentu. Dalam hal ini yang

disebut bentuk normal adalah suatu keadaan relasi yang dihasilkan oleh penerapan aturan-aturan sederhana yang berhubungan dengan dependensi fungsional terhadap relasi tersebut. Gambar II.4 berikut ini akan memperlihatkan hubungan keenam bentuk normal tersebut.



Gambar II.5 Langkah-langkah dalam Normalisasi

(Sumber : Abdul Kadir ; 2009 : 118)

II.6. *Unified Modeling Language* (UML)

Menurut Rosa A.S & M. Shalahuddin (2011 : 118), pada perkembangan teknik pemrograman berorientasi objek, muncullah sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemrograman berorientasi objek, yaitu *Unified Modeling Language* (UML). UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun dan dokumentasi dari

sistem perangkat lunak. UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung.

UML hanya berfungsi untuk melakukan pemodelan, jadi penggunaan UML tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya UML paling banyak digunakan pada metode berorientasi objek.

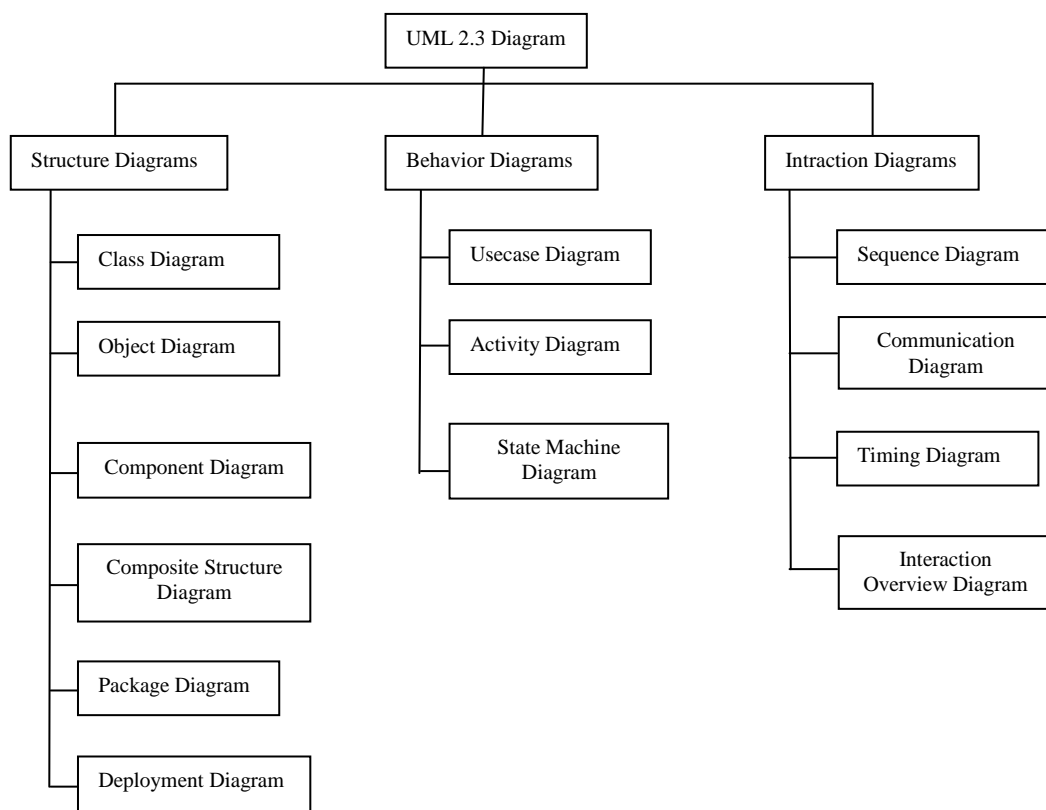
Menurut Prabowo Pudjo Widodo & Herlawati (2011 : 6), UML diaplikasikan untuk maksud tertentu, biasanya antara lain :

1. Merancang perangkat Lunak.
2. Sarana Komunikasi antara perangkat lunak dengan proses bisnis.
3. Menjabarkan sistem secara rinci untuk analisa dan mencari apa yang diperlukan sistem.
4. Mendokumentasikan sistem yang ada, proses-proses dan organisasinya.

Blok pembangunan utama UML adalah diagram. Beberapa diagram ada yang rinci (jenis *timing diagram*) dan lainnya ada yang bersifat umum (misalnya diagram kelas). Para pengembang sistem berorientasi objek menggunakan bahasa model untuk menggambarkan, membangun dan mendokumentasikan sistem yang mereka rancang. UML memungkinkan para anggota team untuk bekerja sama dengan bahasa model yang sama dengan mengaplikasikan beragam sistem. Intinya UML merupakan alat komunikasi yang konsisten dalam mendukung para pengembang sistem saat ini.

II.6.1 Diagram-Diagram UML

Menurut Rosa A.S & M. Shalahuddin (2011 : 120), 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 II.5 di bawah ini



Gambar II.6 Diagram UML

(Sumber : Rosa A.S & M. Shalahuddin ; 2011 : 121)

Berikut ini penjelasan singkat dari pembagian kategori tersebut

1. *StructureDiagrams* 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.

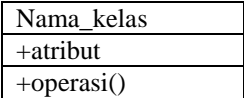
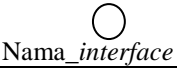

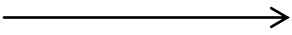
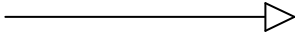

II.6.2. Class Diagram

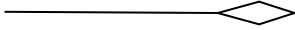
Diagram kelas atau *Class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem.

Kelas memiliki apa yang disebut atribut dan metode atau operasi.

- 1) Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas.
- 2) Operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas.

Tabel II.2. Diagram Kelas

Simbol	Deskripsi
Kelas 	Kelas pada struktur sistem
Antarmuka / <i>interface</i> 	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek
Asosiasi / <i>association</i>  Asosiasi berarah/ <i>directed association</i> 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
Generalisasi 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus)
Kebergantungan 	Relasi antar kelas dengan makna kebergantungan antar kelas


Agregasi / <i>aggregation</i> 	Semua bagian (<i>whole part</i>)
--	------------------------------------

(Sumber : Rosa A.S & M. Shalahuddin ; 2011 : 124)

II.6.3. *Object Diagram*

Diagram objek menggambarkan struktur sistem dari segi penamaan objek dan jalannya objek dalam sistem. Pada diagram objek harus dipastikan semua kelas yang sudah didefinisikan pada diagram kelas harus dipakai objeknya, karena jika tidak, pendefinisian kelas itu tidak dapat dipertanggungjawabkan. Untuk apa mendefinisikan sebuah kelas sedangkan pada jalannya sistem, objeknya tidak pernah dipakai.

Tabel II.3. Diagram Paket

Simbol	Deskripsi		
Objek <table border="1" style="margin-left: 40px;"> <tr> <td style="padding: 2px;">Nama_objek : nama_kelas</td> </tr> <tr> <td style="padding: 2px;">Atribut = nilai</td> </tr> </table>	Nama_objek : nama_kelas	Atribut = nilai	Objek dari kelas yang berjalansaat sistem dijalankan
Nama_objek : nama_kelas			
Atribut = nilai			
Link 	Relasi antar objek		

(Sumber : Rosa A.S & M. Shalahuddin ; 2011 : 124)

II.6.4. *Component Diagram*

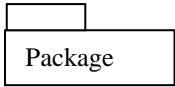
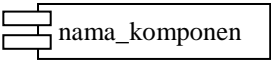
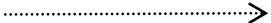


Diagram komponen atau component diagram dibuat untuk menunjukkan organisasi dan ketergantungan di antara kumpulan komponen dalam sebuah sistem. Diagram komponen fokus pada komponen sistem yang dibutuhkan dan

ada didalam sistem. Komponen dasar yang biasanya ada dalam suatu sistem adalah sebagai berikut :

- 1) Komponen *user interface* yang menangani tampilan.
- 2) Komponen *bussiness procesiing* yang menangani fungsi-fungsi proses bisnis.
- 3) Komponen data yang menangani manipulasi data.
- 4) Komponen *security* yang menangani keamanan sistem.

Komponen lebih terfokus pada penggolongan secara umum fungsi-fungsi yang diperlukan, berikut tabel II.4 yang menerangkan simbol-simbol yang ada pada diagram komponen.

Tabel II.4. Diagram Komponen

Simbol	Deskripsi
Package 	<i>Package</i> merupakan sebuah bungkusan dari satu atau lebih komponen
Komponen 	Komponen Sistem
Kebergantungan / <i>dependency</i> 	Kebergantungan antar komponen, arah panah mengarah pada komponen yang dipakai
Antar muka / <i>interface</i> 	Sama dengan konsep <i>interface</i> pada pemrograman berorientasi objek, yaitu sebagai antarmuka komponen agar tidak mengakses langsung komponen
Link 	Relasi antar komponen

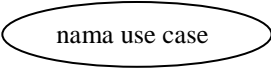
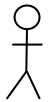

(Sumber : Rosa A.S & M. Shalahuddin ; 2011 : 126)

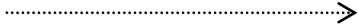
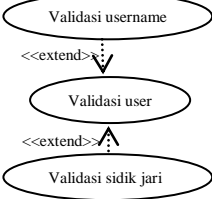
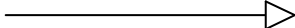
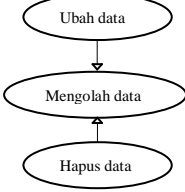
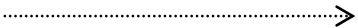
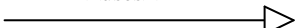
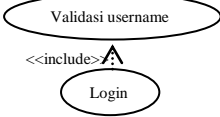
II.6.5. Use Case Diagram

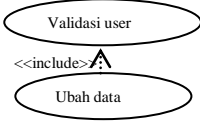
Use case atau diagram *use case* merupakan pemodelan untuk kelakuan (*behaviour*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. 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*.

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

Tabel II.5. Diagram Use Case

Simbol	Deskripsi
Use case  nama use case	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>use case</i>
Aktor / <i>actor</i>  nama aktor	Orang, proses, atau sistem yang lain berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan di buat itu sendiri
Asosiasi / <i>association</i> 	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> , atau <i>usecase</i> memiliki interasi dengan aktor

<p>Ekstensi / <i>extend</i></p> <p style="text-align: center;"><<extend>></p> 	<p>Relasi usecase tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu, mirip dengan prinsip inheritance pada pemrograman berorientasi objek, biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan misal</p>  <p>arah panah mengarah pada <i>use case</i> yang ditambahkan</p>
<p>Generalisasi / <i>generalization</i></p> 	<p>Hubungan generalisasi dan spesialisasi (umum – khusus) antara dua buah use case dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya misalnya :</p>  <p>Arah panah mengarah pada use case yang menjadi generalisasinya (umum)</p>
<p>Menggunakan / <i>include</i> / <i>uses</i></p> <p style="text-align: center;"><<include>></p>  <p style="text-align: center;"><<uses>></p> 	<p>Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan memerlukan use case ini untuk menjalankan fungsinya atau sebagai syarat dijalankan use case ini</p> <p>Ada 2 sudut pandang yang cukup besar mengenai include di usecase</p> <ol style="list-style-type: none"> 1. include berarti use case yang ditambahkan akan selalu dipanggil saat use case dijalankan misal pada kasus berikut : 

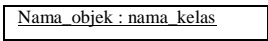

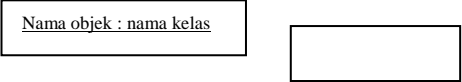
	<p>2. include berarti use case yang tambahan akan selalu melakukan pengecekan apakah use case yang di tambahkan telah di jalankan sebelum use case tambahan di jalankan, misal pada kasus berikut :</p>  <p>Kedua interpretasi di atas dapat dianut salah satu atau keduanya tergantung pada pertimbangan dan interpretasi yang dibutuhkan.</p>
--	---



(Sumber : Rosa A.S & M. Shalahuddin ; 2011 : 131)

II.6.6. Communication Diagram

Diagram komunikasi mengelompokkan message pada kumpulan diagram sekuen menjadi sebuah diagram. Dalam diagram komunikasi yang dituliskan adalah operasi / metode yang di jalankan antara objek yang satu dengan objek lainnya secara keseluruhan, oleh karna itu dapat di ambil dari jalanya interaksi pada semua diagram sekuen. Berikut adalah tabel II.6 yang menerangkan simbol-simbol yang ada pada diagram komunikasi :

Tabel II.6. Diagram Komunikasi

Simbol	Deskripsi
<p>Objek</p> 	<p>Objek yang melakukan interaksi pesan</p>
<p>Link</p> 	<p>Relasi antar objek yang menghubungkan objek satu dengan lainnya atau dengan dirinya sendiri</p> 

Arah pesan / stimulus 	Arah pesan yang terjadi, jika pada suatu link ada dua arah pesan yang berbeda, maka arah juga digambarkan dua arah pada dua sisi link 
--	---

(Sumber : Rosa A.S & M. Shalahuddin ; 2011 : 140)


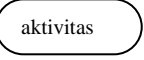


II.6.7. Activity Diagram


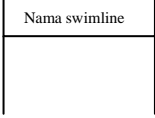

Diagram aktivitas atau activity diagram menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Diagram aktivitas juga banyak digunakan untuk mendefinisikan hal-hal berikut :

- 1) Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.
- 2) Urutan atau pengelompokan tampilan dari sistem/user interface dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan.
- 3) Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujiannya.

Berikut adalah tabel II.7 yang menggambarkan simbol-simbol yang ada pada diagram aktivitas :

Tabel II.7. Diagram Aktivitas

Simbol	Deskripsi
Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki status awal
Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
Percabangan / decision 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
Penggabungan / join 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu

Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
Swimlane  atau 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi


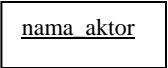
(Sumber : Rosa A.S & M. Shalauddin ; 2011 : 134)


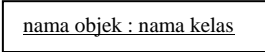

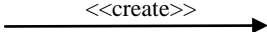
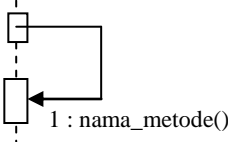
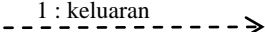
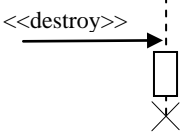
II.6.8. Sequence Diagram

Diagram sekuen menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan message yang dikirimkan dan diterima antar objek. Banyaknya diagram objek yang digambarkan adalah sebanyak pendefinisian use case yang memiliki proses sendiri atau yang penting semua use case yang telah didefinisikan interaksi jalanya pesan sudah dicakup pada diagram sekuen sehingga semakin banyak use case yang didefinisikan maka diagram sekuen yang harus dibuat juga semakin banyak.

Berikut adalah tabel II.8 yang menerangkan simbol-simbol yang ada pada diagram sekuen :

Tabel II.8. Diagram Squence

Simbol	Deskripsi
Aktor  nama aktor atau  tampa waktu aktif	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, biasanya di nyatakan menggunakan kata benda di awali frase nama aktor

<p>Garis hidup / lifeline</p> 	<p>Menyatakan kehidupan suatu objek</p>
<p>Objek</p> 	<p>Menyatakan objek yang berinteraksi pesan</p>
<p>Waktu aktif</p> 	<p>Menyatakan objek dalam keadaan aktif dan berinteraksi pesan</p>
<p>Pesan tipe create</p> 	<p>Objek yang lain, arah panah mengarah pada objek yang dibuat</p>
<p>Pesan tipe call</p>	<p>Menyatakan suatu objek memanggil operasi / metode yang ada pada objek lain atau dirinya sendiri</p>  <p>Arah panah mengarah pada objek yang memiliki operasi / metode, karena ini memanggil operasi / metode maka operasi / metode yang di panggil harus ada pada diagram kelas sesuai dengan kelas objek yang berinteraksi</p>
<p>Pesan tipe send</p>	<p>Menyatakan bahwa suatu objek mengirimkan data / masukan / informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim</p>
<p>Pesan tipe return</p> 	<p>Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian</p>
<p>Pesan tipe destroy</p> 	<p>Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada create maka ada destroy</p>

(Sumber : Rosa A.S & M. Shalahuddin ; 2011 : 138)

II.7. Pemrograman Visual Basic 2010

Menurut Wahana Komputer (2012 : 7) Visual Basic 2010. NET merupakan sebuah *Integrated Development Environment* (IDE) atau lingkungan kerja yang digunakan untuk membangun aplikasi. NET dengan mudah. Visual Studio Profesional 2010 menyediakan berbagai tool yang lengkap bagi para pengembang untuk membangun aplikasi yang berjalan di . NET Framework. Berbagai tool, antara lain tool Toolbox yang berisi komponen visual, sehingga anda tinggal drag and drop komponen dan *Visual Studio 2010* akan menuliskan kode untuk anda. Selain itu masih ada jendela *wizard* yang membantu anda untuk melakukan pemrograman dengan sangat mudah.

Selain itu anda juga dapat menulis kode Visual Studio 2010 pada lingkungan kerja lain, seperti Visual basic Express Edition yang juga disediakan oleh Microsoft secara *free* bagi para pelajar dan pemula. Pada *Visual Basic Express Edition*, tool-tool yang terpasang sangat minim dan tidak memiliki komponen visual sehingga anda harus menuliskan kode dari awal. Anda juga dapat menuliskan kode pada editor yang sangat sederhana, yaitu *notepad*.