

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **II.1. Sistem**

Sistem sangat dibutuhkan dalam mengolah data yang mengalir didalamnya maupun diluar lingkungan instansi ataupun perusahaan. Sistem harus dirancang sedemikian rupa agar dapat menentukan validitas data yang berasal dari berbagai sumber seperti konsumen ataupun kelompok lainnya. Suatu sistem dalam mencapai tujuannya tentu terdiri dari bermacam-macam proses baik secara fisik, konsep maupun secara prosedural.

Sistem merupakan sekelompok unsur yang erat hubungannya satu dengan yang lainnya, yang berfungsi bersama-sama untuk mencapai tujuan tertentu (Tata Sutabri : 2004 ; 3).

Sistem adalah sekumpulan elemen yang saling terkait atau terpadu yang dimaksudkan untuk mencapai suatu tujuan (Abdul Kadir : 2003 ; 54 ).

Maka dari beberapa definisi sistem diatas penulis menarik kesimpulan bahwa sistem merupakan suatu kumpulan elemen dan komponen yang secara bersama-sama membentuk suatu fungsi yang sama untuk mencapai tujuan yang ditetapkan.

## **II.2. Informasi**

Informasi adalah data yang telah diolah menjadi sebuah bentuk yang berarti bagi penerimanya dan bermanfaat dalam pengambilan keputusan saat ini atau saat mendatang (Abdul kadir : 2003 ; 31).

Informasi adalah data yang telah diklasifikasikan atau diolah atau diinterpretasikan untuk digunakan dalam proses pengambilan keputusan (Tata Sutabri : 2004 ; 18).

Maka informasi merupakan data yang telah diolah dan bermanfaat bagi yang menerima yang berisikan fakta-fakta yang mudah untuk dipahami oleh orang agar suatu informasi dapat berkualitas yaitu akurat, tepat waktu dan relevan.

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

Sistem informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan. (Tata Sutabri : 2004 ; 36).

Menurut Wilkinson (1992) Sistem Informasi adalah kerangka kerja yang mengkoordinasikan sumber daya (manusia, komputer) untuk mengubah masukan (input) menjadi keluaran (informasi), guna mencapai sasaran-sasaran perusahaan. (Abdul kadir : 2003 ; 11).

Dari penjelasan mengenai sistem dan informasi dapat disimpulkan bahwa sistem informasi merupakan suatu elemen yang saling berhubungan untuk menyediakan suatu informasi yang digunakan untuk pengambilan keputusan

dalam manajemen operasional dan perusahaan dalam pengambilan keputusan dan memberikan data yang akurat, tepat waktu dan dapat dipahami oleh orang yang menerima dan membacanya. Ada beberapa tujuan sistem informasi yaitu :

1. Menyediakan informasi untuk membantu pengambilan keputusan manajemen.
2. Membantu petugas dalam melaksanakan tugas operasional perusahaan dari hari ke hari.
3. Menyediakan informasi yang layak bagi pemakai pihak luar perusahaan.

#### **II.4. Pemasaran**

Pemasaran merupakan suatu aktivitas dimana perusahaan menawarkan suatu barang dan jasa ke berbagai kelompok sosial untuk memenuhi kebutuhannya. Pemasaran juga sebagai kegiatan manusia diarahkan untuk memuaskan keinginan dan kebutuhan melalui proses pertukaran.

Pemasaran menurut Philip Kotler (2009 : 5) Pemasaran adalah suatu fungsi organisasi dan serangkaian proses untuk menciptakan, mengkomunikasikan dan memberikan nilai kepada pelanggan dan untuk mengelola hubungan pelanggan dengan cara menguntungkan organisasi dan pemangku kepentingannya.

Fungsi pemasaran sangatlah penting dalam memasarkan suatu produk baik barang ataupun jasa, karena diperlukan strategi-strategi dalam menetapkan harga dan mempromosikan barang atau jasa. Pemasaran bertujuan untuk memenuhi kebutuhan, keinginan dan permintaan akan barang dan jasa yang diperlukan oleh pasar. Adapun strategi pemasaran yaitu :

- a. Mempelajari kebutuhan dan keinginan pelanggan dalam segmen pasar yang ditentukan dengan cermat.
- b. Mengukur citra perusahaan dan kepuasan pelanggan secara terus-menerus.
- c. Mengumpulkan dan mengevaluasi ide-ide produk baru,perbaikan produk dan pelayanan.
- d. Mengembangkan penawaran yang memikat untuk setiap segmen sasaran.

(Sumber : Philip Kotler, Kevin Lane Keller : 2009 ; 27).

#### **II.4.1. Pemasaran online**

Pemasaran internet atau e-pemasaran/online pemasaran merupakan pemasaran suatu

barang dan jasa menggunakan media internet atau jaringan. Pemasaran secara online merupakan kegiatan pemasaran yang meliputi pembuatan desain web, periklanan dan promosi barang atau jasa lewat media internet.

Pemasaran produk kuliner pada CV.kiki Anugrah masih menggunakan cara yang manual. Oleh karena itu dengan dirancangnya suatu media berbasis website dapat memudahkan dalam memasarkan produk kuliner.

Dengan demikian penulis mencoba untuk membuat sistem pemasaran produk kuliner berbasis web yang menyediakan jasa pemesanan produk kuliner secara online pada CV.KikiAnugrah yang nantinya akan membantu kebutuhan masyarakat.

## II.5. Barang

Barang merupakan suatu produk fisik yang berwujud yang dapat diberikan kepada seseorang pembeli dan melibatkan perpindahan kepemilikan dari penjual ke pelanggan. (Sumber : Philip Kotler, Kevin Lane Keller : 2009 ; 6).

## II.6. PHP

PHP (*Hypertext Preprocessor*) merupakan skrip yang berjalan dalam *server side* yang ditambahkan dalam HTML. Skrip ini akan membuat suatu aplikasi dapat diintegrasikan kedalam HTML sehingga suatu halaman HTML tidak lagi bersifat *statis*, namun menjadi bersifat *dinamis*. Sifat *server side* ini membuat pengerjaan skrip tersebut dikerjakan *diserver* sedangkan yang dikirimkan kepada *browser* adalah hasil proses dari skrip tersebut yang sudah berbentuk HTML.

PHP dirancang untuk membentuk aplikasi web dinamis. Artinya, PHP dapat membentuk suatu tampilan berdasarkan permintaan. Pada prinsipnya PHP mempunyai fungsi yang sama dengan skrip-skrip seperti ASP (*Active Server Page*), *Cold Fusion*, ataupun *Perl*. Namun perlu diketahui bahwa PHP sebenarnya bisa dipakai secara *Command Line*, artinya skrip PHP dapat dijalankan tanpa melibatkan *web server* maupun *browser*. (Sumber : Abdul Kadir : 2008 ; 2)

## II.7. Kelebihan PHP

Ada beberapa kelebihan yang dimiliki oleh PHP yaitu :

- a. Bahasa pemrograman PHP adalah sebuah bahasa skrip yang tidak melakukan sebuah kompilasi dalam penggunaannya.

- b. *Web server* mendukung PHP dapat ditemukan dimana-mana dari mulai *apache,IIS* hingga *web server* yang lain dengan konfigurasi yang relative mudah.
- c. PHP adalah bahasa open source yang dapat digunakan diberbagai mesin (*Linux,Unix,Windows*) dan dapat dijalankan secara *runtime* melalui *cencole* serta juga dapat menjalankan perintah-perintah sistem.
- d. PHP mampu berkomunikasi dengan berbagai database yang terkenal meskipun dengan kelengkapan yang berbeda-beda,sehingga PHP sangat cocok untuk membangun halaman-halaman *web dinamis*.

(Sumber : Bunafit Nugroho : 2004 ; 140-141).

### **II.7.1. Konsep Dasar PHP**

PHP adalah bahasa skrip yang menyatu dengan HTML dan dijalankan pada server side. Artinya semua sintaks yang kita berikan akan sepenuhnya dijalankan pada server, sedangkan yang dikirim ke browser hanya hasilnya saja.

Untuk menuliskan dan memperkenalkan PHP dimulai dengan tanda `<?php`, setelah tanda tersebut dilanjutkan dengan kode program didalamnya. Untuk mengakhiri kode program tanda seperti itu, dapat juga digunakan beberapa bentuk lain yaitu sebagai berikut : (Sumber : Bunafit Nugroho : 2004 ; 144)

**Tabel II.1 Syntax Dasar PHP**

Awal	Akhir
< ?	? >
< ? php	? >
<script language = "php"	</script>
<%	%>

( Sumber : Bunafit Nugroho : 2004 ; 144 )

### II.7.2. Konsep Kerja PHP

Model kerja HTML diawali dengan permintaan satu halaman web oleh *browser*. Berdasarkan URL (*Uniform Resource Locator*) atau dikenal dengan sebutan alamat internet, *browser* mendapatkan alamat dari *web server*, mengidentifikasi halaman yang diinginkan, dan menyampaikan segala informasi yang dibutuhkan oleh *web server*.

Selanjutnya *web server* akan mencari *file* yang diminta dan diproses. Setelah diproses maka web server akan memberikan isinya ke *web browser* dan *browser* yang mendapatkan isinya segera melakukan proses penerjemahan kode HTML dan menampilkannya ke layar pemakai. ( Sumber : Abdul Kadir : 2009 ; 5)

### II.8. MySQL

*MySQL* (*My Structure Query Language*) adalah sebuah program pembuatan database yang bersifat *open-source*, artinya siapa saja boleh menggunakannya. *MySQL* sebenarnya produk yang berjalan pada *platform Linux*. Karena sifatnya yang *open-source*, *MySQL* dapat dijalankan pada semua *platform*

baik *Windows* maupun *Linux*. Selain itu *MySQL* juga merupakan program pengakses *database* yang bersifat jaringan sehingga dapat digunakan untuk aplikasi *Multi User* (Banyak Pengguna).

Kelebihan lain dari *MySQL* adalah menggunakan bahasa Query standar yang dimiliki *SQL* (*Structure Query Language*). *SQL* adalah suatu bahasa permintaan yang terstruktur yang telah distandarkan untuk semua program pengakses *database*. *MySQL* dapat didukung oleh hampir semua program aplikasi baik open source seperti *PHP* karena kedekatannya terhadap aplikasi web. (Sumber : Bunafit Nugroho : 2004 ; 26).

Dengan demikian dalam memanipulasi data penulis menggunakan *MySQL* sebagai *database* dalam pengolahan data.



**Gambar II.1. MySQL diakses dengan phpMyAdmin**

( Sumber : Abdul Kadir : 2009 ; 17 )



## II.9. Entity Relationship Diagram (ERD)

Merupakan suatu model untuk menjelaskan hubungan antar dua dalam basis data berdasarkan suatu persepsi bahwa *real word* terdiri dari *object-object* dasar yang mempunyai hubungan atau antar *object-object* tersebut. Relasi antar *object* dengan menggunakan symbol-simbol grafis tertentu. Model *entity relationship* adalah suatu penyajian dengan menggunakan *entity* dan *relationship*. Diperkenalkan pada tahun 1976 oleh P.P. Chen.

### II.9.1. Komponen-komponen yang terdapat didalam Entity Relationship Model.

#### 1. Entity

- a. Adalah sesuatu yang dapat dibedakan dalam dunia nyata dimana informasi yang berkaitan dengannya dikumpulkan.
- b. *Entity* set adalah kumpulan *entity* yang sejenis.
- c. Symbol yang digunakan untuk *entity* adalah persegi panjang.
- d. *Entity* set dapat berupa :
  1. *Entity* yang bersifat fisik, yaitu *entity* yang dapat dilihat.  
Contohnya : rumah, kendaraan, mahasiswa, dosen, dan lain-lain.
  2. *Entity* yang bersifat konsep atau logic, yaitu *entity* yang tidak dapat dilihat.  
Contohnya : pekerjaan, perusahaan, rencana, mata kuliah, dan lain-lain.
- e. Simbol yang digunakan untuk *entity* adalah persegi panjang.

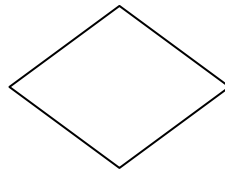


Gambar II.2. Entity

( Sumber : Linda Marlinda, S. Kom; 2004:17 )

## 2. Relationship

- a. Adalah hubungan yang terjadi antara satu atau lebih *entity*.
- b. *Relationship* tidak mempunyai keberadaan fisik, kecuali yang mewarisi hubungan antara *entity* tersebut.
- c. *Relationship* set adalah kumpulan relationship yang sejenis.
- d. Simbol yang digunakan adalah bentuk belah ketupat, *diamond* atau *rectangle*.

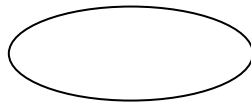


**Gambar II.3. Relationship**

( Sumber : Linda Marlinda, S. Kom; 2004:18 )

## 3. Attribute

- a. Adalah karakteristik dari *entity* atau *relationship* yang menyediakan penjelasan detail tentang atau *relationship* tersebut.
- b. Attribute value (nilai atribut) adalah suatu data aktual atau informasi yang disimpan di suatu atribut di dalam suatu *entity* atau *relationship*.
- c. Terdapat dua jenis atribut, yaitu :
  1. *Indetifer (key)*, untuk menentukan suatu *entity* secara unik.
  2. *Descriptor (nonkey attribute)*, untuk menentukan karakteristik dari suatu *entity* yang tidak unik.
- d. Simbol yang digunakan adalah bentuk oval



**Gambar II.4. Attribute**

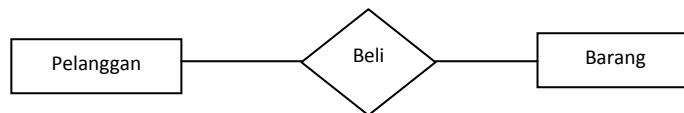
( Sumber : Linda Marlinda, S. Kom; 2004:18 )

#### 4. Indicator Tipe

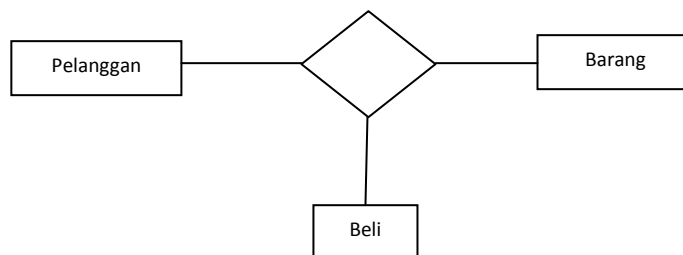
##### a. *Indicator tipe associative object*

Berfungsi sebagai suatu objek dan suatu *relationship*

Contoh :



Menjadi :



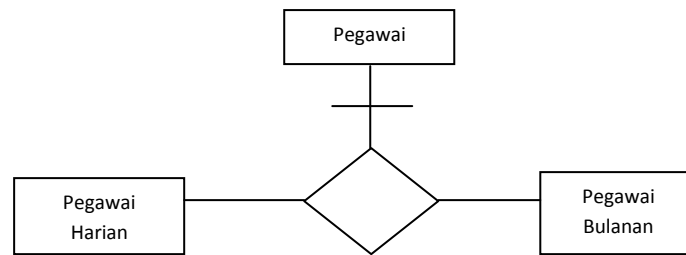
**Gambar II.5. Indicator Tipe**

( Sumber : Linda Marlinda, S. Kom; 2004:19 )

##### b. Indicator tipe supertipe.

Terdiri dari suatu object dan sub kategori atau lebih yang dihubungkan dengan dihubungkan dengan relationship yang tidak bernama (Linda Marlinda, S. Kom 2004:17-19).

Contoh :



**Gambar II.6. Indikator Tipe SuperTipe**

( Sumber : Linda Marlinda, S. Kom; 2004:19 )

## II.10. Normalisasi

Normalisasi adalah proses pengelompokkan elemen data menjadi tabel-tabel yang menunjukkan entitas dan relasinya.

Ada beberapa macam kunci (*key function*) yang digunakan untuk proses pencarian, penyaringan, penghapusan, dan lainnya yaitu sebagai berikut :

1. *Candidate Key* (kunci Calon).

Kunci kandidat adalah satu atribut atau satu set minimal atribut yang mengidentifikasi secara unik suatu kejadian yang spesifik dari suatu entitas.

2. *Primary Key* ( Kunci primer).

Kunci primer adalah satu atribut atau satu set minimal atribut yang tidak hanya mengidentifikasi secara unik suatu kejadian yang spesifik, akan tetapi juga dapat mewakili setiap kejadian dari suatu entitas. Setiap kunci kandidat punya peluang menjadi *primary key*, akan tetapi sebaiknya dipilih satu saja yang dapat mewakili entitas yang ada secara menyeluruh.

3. *Alternate Key* (Kunci alternatif).

Kunci alternatif adalah kunci kandidat yang tidak dipakai sebagai *primary key*. Kerap kali kunci alternatif ini dipakai sebagai kunci pengurutan dalam pembuatan laporan.

4. *Foreign Key* (Kunci tamu)

Kunci tamu adalah satu atribut atau satu set atribut yang melengkapi satu *relationship* (hubungan) yang menunjukkan ke induknya.

Pada proses normalisasi ini perlu dikenal dahulu definisi dari tahap normalisasi, yaitu sebagai berikut :

1. Bentuk Tidak Normal (*Unnormalized Form*).

Bentuk ini merupakan kumpulan data yang akan direkam, tidak ada keharusan untuk mengikuti suatu format tertentu, dapat saja data tidak lengkap atau terduplikasi. Data dikumpulkan apa adanya sesuai dengan kedatangannya.

2. Bentuk Normal Kesatu (*1NF/First Normal Form*).

Bentuk normal kesatu mempunyai ciri yaitu bahwa setiap data dibentuk dalam *flat file* (file datar/rata), data dibentuk dalam satu *record* demi *record* dan nilai dari *field* berupa *atomic value*. Tidak ada set atribut yang berulang atau atribut bernilai ganda (*multivalue*).

3. Bentuk Normal Kedua (*2NF/Second Normal Form*).

Bentuk normal kedua mempunyai syarat, yaitu bentuk data telah memenuhi kriteria bentuk normal kesatu. Atribut bukan kunci haruslah bergantung secara fungsi pada kunci utama / *primary key*. Dengan

demikian untuk membentuk normal kedua haruslah sudah ditentukan kunci *field*, kunci *field* haruslah unik dan dapat mewakili atribut lain yang menjadi anggotanya.

4. Bentuk Normal Ketiga (3NF/*Third Normal Form*).

Untuk menjadi bentuk normal ketiga maka relasi haruslah dalam bentuk normal kedua dan semua atribut bukan primer tidak punya hubungan yang transitif. Dengan kata lain, setiap atribut bukan kunci haruslah bergantung hanya pada *primary key* secara menyeluruh.

5. *Boyce-Codd Normal Form* (BCNF).

*Boyce-Codd Normal Form* mempunyai paksaan yang lebih kuat dari bentuk normal ketiga. Untuk menjadi BCNF, relasi harus dalam bentuk normal kesatu dan setiap atribut harus bergantung fungsi pada atribut *superkey*. (Tata Sutabri : 2004 ; 202-208).

### II.10.1. Kamus Data

Kamus data (KD) atau data *dictionary* (DD) atau disebut juga dengan istilah *systems data dictionary* adalah katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi.

Dengan menggunakan KD, analisis sistem dapat mendefinisikan data yang mengalir disistem dengan lengkap. KD dibuat pada tahap perancangan sistem dan digunakan baik pada tahap analisis maupun pada tahap perancangan sistem. Pada tahap analisis, KD dapat digunakan sebagai alat komunikasi antara analisis sistem dengan pemakai sistem tentang data yang mengalir disistem, yaitu tentang data yang masuk kesistem dan tentang informasi yang dibutuhkan oleh pemakai

sistem. Pada tahap perancangan sistem KD digunakan untuk merancang input, merancang laporan-laporan dan database. KD dibuat berdasarkan arus data yang ada di DAD. Arus data di DAD sifatnya global, hanya ditunjukkan nama arus datanya saja (Prof. Dr. Jogiyanto HM, MBA, Akt : 2005 ; 725).

**Tabel II.2. Simbol-simbol dalam kamus data**

<b>Simbol</b>	<b>Uraian</b>
=	Terdiri atas, mendefinisikan, diuraikan menjadi, artinya contoh : nama = sebutan + nama1 + nama2 + gelar1 + gelar2
+	Dan
()	Optional (pilihan boleh ada atau boleh tidak) Contoh : alamat = alamat rumah + (alamat surat)
{ }	Pengulangan Contoh : nama1 = {karakter_valid}
[ ]	Memilih salah satu dari sejumlah alternative, seleksi Contoh : sebutan = [Bapak Ibu Yang Mulia]
* *	Komentar Contoh : *seminar yang akan diikuti*
	Pemisah sejumlah alternatif pilihan antara simbol [ ]

( Sumber : Budi Sutedjo Dharma Oetomo, S.Kom., MM 2006 :118-119 )

Berikut ini contoh penggunaan kamus data untuk perancangan database pembicara seminar :

1. Data\_pembicara=kode\_pembicara+nama+alamat+telepon+tanggal\_lahir  
+honor+spesialisasi
2. Kode\_pembicara=level\_pembicara+nomor\_urut
3. Nama=sebutan+nama1+nama2+gelar1+gelar2+gelar3
4. Alamat=nama\_jalan+nomor+kota+Negara+kodepos
5. Telepon+telepon\_rumah+ponsel
6. Tanggal\_lahir=\*dd/mm/yyyy\*
7. Honor=\*99.999.999\*
8. Spesialisasi=0{karakter}15

### **II.10.2. Macromedia Dreamweaver**

Macromedia DreamWeaver adalah sebuah software web design yang menawarkan cara mendesain web dengan dua langkah sekaligus dalam satu waktu,yaitu mendesain dan memprogram. DreamWeaver memiliki satu jendela mini yang disebut HTML source tempat kode-kode HTML tertulis.

(Sumber : Gregorius Agung : 2001 ; 9).

Macromedia DreamWeaver merupakan program editor HTML visual yang paling sering digunakan untuk mengelola situs dan menata layout halaman web,karena dalam pembuatannya tidak perlu mengetikkan syntax-syntax HTML. Dengan megalami perubahan yang akan warna dan area kerjanya menjadi lebih ringkas dan efesien Macromedia DreamWeaver banyak digunakan oleh para



progremers dan pemula. Dalam pembuatan website pada Perum Pegadaian untuk memasarkan barang lelang penulis menggunakan Macromedia DreamWeaver 8.



**Gambar II.7. Macromedia DreamWeaver**

( Sumber : Ali Akbar,ST : 2006 ; 12 )

### II.10.3. Aplikasi Web

Kepopuleran internet di seluruh penjuru dunia mendorong aplikasi web semakin diminati. Dengan menggunakan aplikasi web hanya perlu menempatkan aplikasi dalam sebuah server dan dengan sendirinya aplikasi tersebut dapat diakses dari mana pun, sepanjang pemakai dapat mengakses *web servernya*. Aplikasi web yang paling dasar ditulis dengan menggunakan HTML. Sebagaimana diketahui, HTML (*Hypertext Markup Language*) adalah bahasa standar untuk membuat halaman-halaman web. ( Sumber : Abdul Kadir : 2009 ; 2)

### II.10.4. Web Browser

Web browser adalah software yang digunakan untuk menampilkan informasi dari server web. Ada empat web browser GUI (*Graphical User*

*Interface*) yang populer yaitu : Internet Explorer, Netscape Navigator, Opera dan Mozilla. (Sumber : Abdul Kadir : 2009 ; 2).

#### **II.10.5. Web Server**

Web server adalah komputer yang digunakan untuk menyimpan dokumen-dokumen web, komputer ini akan melayani permintaan dokumen web dari kliennya. Contoh web server adalah :

1. Apache, merupakan web server yang digunakan diberbagai platform *Operting System (OS)* ,seperti Linux, windows dan lain-lain.
2. IIS (*Internet Information Service*), digunakan di system operasi Windows NI dan Windows 2002.
3. PWS (*Presonal Web server*), digunakan disytem operasi windows 9x.

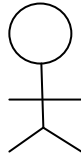
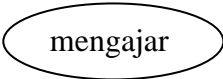
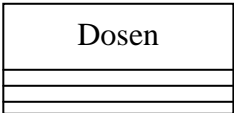
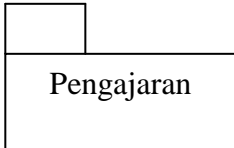
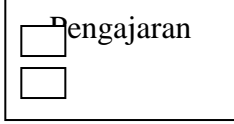
(Sumber : Bunafit Nugroho : 2004 ; 6).


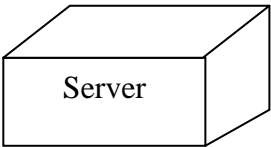
#### **II.11. UML (Unified Modeling Language)**

UML (Unified Modeling Language) adalah bahasa pemodelan untuk sistem atau perangkat lunak yang berparadigma berorientasi objek. Pemodelan (modeling) digunakan untuk penyederhanaan permasalahan-permasalahan yang kompleks sedemikian rupa sehingga lebih mudah dipelajari dan dipahami.

.(Sumber : Adi Nugroho : 2010 ; 6)


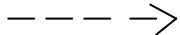
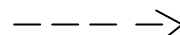
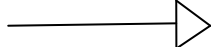
**Tabel II.3. Beberapa Pengklasifikasi (Classifier)**

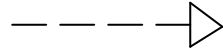
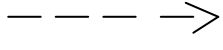
Pengklafikasi	Kegunaan	Notasi
Actor	Menggambarkan semua objek diluar sistem(bukan hanya pengguna sistem/perangkat lunak) yang berinteraksi dengan sistem yang digunakan.	
Use case	Menggambarkan fungsionalitas yang dimiliki sistem.	
Kelas (Class)	Menggambarkan konsep dasar pemodelan sistem.	
Subsistem (Subsystem)	Menggambarkan paket spesifikasi serta implementasi.	
Komponen (Component)	Menggambarkan bagian-bagian fisik sistem/perangkat lunak yang dikembangkan.	

Antar muka(Interface)	Menggambarkan antar muka pengiriman pesan (message) antar pengklafikasi.	 Antarmuka pengajar-mahasiswa
Simpul (node)	Menggabarkan sumber daya komputasional yang digunakan oleh sistem.	 Server

( Sumber : Adi Nugroho : 2010 ; 16 )

**Tabel II.4. Relasi-Realasi dalam UML**

Relasi	Fungsi	Notasi
Asosiasi (Association)	Mendeskripsikan hubungan antar- instance suatu kelas	
Kebergantungan (Dependency)	Relasi antardua elemen model	
Aliran (flow)	Relasi antardua versi suatu objek	
Generalisasi (Generalization)	Relasi antara pengklasifikasi yang memiliki deskripsi yang bersifat lebih umum dengan berbagai pengklasifikasi yang lebih spesifik, digunakan dalam struktur pewarisan.	

Realisasi (realization)	Relasi antara spesifikasi dan implementasinya.	
Penggunaan (usage)	Situasi dimana salah satu elemen membutuhkan elemen yang lainnya agar dapat berfungsi dengan baik.	

( Sumber : Adi Nugroho :2010 ; 23 )

### II.11.1 Diagram-Diagram *UML*

Beberapa literatur menyebutkan bahwa *UML* menyediakan Sembilan jenis diagram, yang lain menyebutkan delapan karena ada beberapa yang digabung, misalnya diagram komunikasi, diagram urutan, dan diagram pewaktuan digabung menjadi diagram interaksi. Namun demikian model-model itu dapat dikelompokkan berdasarkan sifatnya yaitu statis atau dinamis. Jenis diagram itu antara lain :

1. Diagram Kelas. Bersifat statis. Diagram ini memperlihatkan himpunan kelas-kelas, antarmuka-antarmuka, kolaborasi, serta relasi-relasi diagram. Diagram ini umu dijumpai pada pemodelan sistem berorientasi objek. Meskipun bersifat statis, sering pula diagram kelas memuat kelas-kelas.
2. Diagram paket (*Package Diagram*) bersifat statis. Diagram ini memperlihatkan kumpulan kelas-kelas merupakan bagian dari diagram komponen.

3. Diagram *Use Case* bersifat statis. Diagram ini memperlihatkan himpunan *use-case* dan aktor-aktor (suatu jenis khusus dari kelas). Diagram ini terutama sangat penting untuk mengorganisasi dan memodelkan perilaku suatu sistem yang dibutuhkan serta diharapkan pengguna.
4. Diagram interaksi dan *Sequence* (urutan). Bersifat dinamis. Diagram urutan adalah diagram interaksi yang menekankan pada pengiriman pesan dalam waktu tertentu.
5. Diagram komunikasi (*Communication Diagram*) bersifat dinamis. Diagram sebagai pengganti diagram kolaborasi *UML* yang menekankan organisasi *structural* dari objek-objek yang menerima serta mengirim pesan.
6. Diagram *Statechart* (*Statechart Diagram*) bersifat dinamis. Diagram status memperlihatkan keadaan-keadaan pada sistem, memuat status (*State*), transisi kejadian serta aktifitas. Diagram ini terutama penting untuk memperlihatkan sifat dinamis dari antarmuka (*interface*), kelas, kolaborasi dan terutama penting pada pemodelan sistem-sistem yang reaktif.
7. Diagram aktivitas (*Activity Diagram*) bersifat dinamis. Diagram aktivitas adalah tipe khusus dari diagram status yang memperlihatkan aliran dari suatu sistem. Diagram ini terutama penting dalam pemodelan fungsi-fungsi suatu sistem dan member tekanan pada aliran kendali antar objek.
8. Diagram komponen (*Component Diagram*) bersifat statis. Diagram komponen ini memperlihatkan organisasi serta kebergantungan sistem/perangkat lunak pada komponen-komponen yang telah ada

sebelumnya. Diagram ini berhubungan diagram kelas dimana komponen dipetakan kedalam satu atau lebih kelas-kelas. Antarmuka-antarmuka serta kolaborasi-kolaborasi.

9. Diagram *Deployment* (*Deployment Diagram*) bersifat statis. Diagram ini memperlihatkan konfigurasi saat aplikasi dijalankan (*run time*). Memuat simpul-simpul beserta komponen-komponen yang ada di dalamnya. Diagram *Deployment* berhubungan erat dengan diagram komponen dimana diagram ini memuat satu atau lebih komponen-komponen. Diagram ini sangat berguna saat aplikasi kita berlaku sebagai aplikasi yang dijalankan pada banyak mesin (*distributed computing*). (Sumber : Prabowo Pudji Widodo, Herlawati : 2011 ; 10-12).