

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Konsep Dasar Sistem Informasi**

Sistem informasi dapat didefinisikan sebagai suatu sistem di dalam organisasi yang merupakan kombinasi dari orang-orang, fasilitas, teknologi, media, prosedur-prosedur dan pengendalian yang ditujukan untuk mendapatkan jalur komunikasi penting memproses tipe transaksi rutin tertentu, memberi sinyal kepada manajemen dan yang lainnya terhadap kejadian-kejadian internal dan eksternal yang penting dan menyediakan suatu dasar informasi untuk pengambilan keputusan (Jogiyanto, 2005).

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

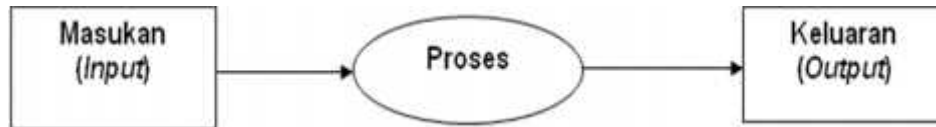
Secara umum, sistem dapat didefinisikan sebagai sekumpulan hal atau kegiatan atau elemen atau subsistem yang saling bekerja sama atau yang dihubungkan dengan cara-cara tertentu sehingga membentuk satu kesatuan untuk melaksanakan suatu fungsi guna mencapai suatu tujuan (Edhy Sutanta, 2003). Disamping sistem dapat didefinisikan sebagai suatu komponen atau subsistem yang berinteraksi untuk mencapai tujuan (Jogiyanto, 2005).

Karakteristik sistem terdiri dari:

1. Mempunyai komponen (*componens*)
2. Mempunyai batas (*boundary*)
3. Mempunyai lingkungan (*environment*)
4. Mempunyai penghubung antar muka (*interface*) antar komponen
5. Mempunyai masukan (*input*)
6. Mempunyai Pengolahan (*processing*)
7. Mempunyai keluaran (*output*)
8. Mempunyai sasaran (*objectives*) dan tujuan (*goal*)
9. Mempunyai kendali (*control*)

10. Mempunyai umpan balik (*feed back*)

Model umum sistem adalah terdiri atas masukan (*input*), pengolahan (*processing*) dan keluaran (*output*), sebagaimana yang ditunjukkan pada gambar di bawah ini.



Gambar 2.1 Model Umum Suatu Sistem (Sumber: Jogiyanto,2005)

### 2.1.2 Pengertian Informasi

Informasi merupakan hasil pengolahan data sehingga menjadi bentuk yang penting bagi penerimaannya dan mempunyai kegunaan sebagai dasar dalam pengambilan keputusan yang dapat dirasakan akibatnya secara langsung saat itu juga atau tidak langsung pada saat mendatang (Edhy Sutanta, 2003,).

Atau informasi adalah proses lebih lanjut dari data dan memiliki nilai tambah.

Syarat informasi yang berkualitas adalah:

1. Akurat, Adalah informasi yang bebas dari kesalahan, tidak bisa dan tidak menyesatkan.
2. Tepat waktu, Adalah informasi yang datang pada penerima tidak boleh terlambat
3. Relevan, Bahwa informasi tersebut mempunyai manfaat untuk pemakai.

### 2.1.3 Pengertian Sistem Informasi

Sistem informasi dapat didefinisikan sebagai suatu sistem di dalam organisasi yang merupakan kombinasi dari orang-orang, fasilitas, teknologi, media, prosedur-prosedur dan pengendalian yang ditujukan untuk mendapatkan jalur komunikasi penting memproses tipe transaksi rutin tertentu, memberi sinyal kepada manajemen dan yang lainnya terhadap kejadian-kejadian internal dan eksternal yang penting dan menyediakan suatu dasar informasi untuk pengambilan keputusan (Jogiyanto, 2005).

Sistem informasi menurut Robert A. Leitch dan K. Roscoe Davis (*Accounting Information System*) 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”.

## **2.2 Perlunya Pengembangan Sistem**

Pengembangan sistem informasi (*information system development*) dapat berarti menyusun suatu sistem yang baru untuk menggantikan sistem yang lama secara keseluruhan atau memperbaiki sistem yang telah ada. (Jogiyanto, 2005).

### **2.2.1. Alasan Pengembangan Sistem Informasi**

Ada beberapa alasan mengapa sistem informasi yang lama perlu diperbaiki atau diganti, yaitu :

1. Adanya permasalahan-permasalahan (*problems*) yang timbul pada sistem informasi lama. Permasalahan-permasalahan yang timbul berupa:
  - 1) Ketidakberesan
    - a. Kecurangan-kecurangan yang tidak disengaja yang menyebabkan tidak amannya harta kekayaan perusahaan.
    - b. Tidak efisiennya operasi.
  - 2) Pertumbuhan Organisasi  
Pertumbuhan organisasi diataranya kebutuhan informasi yang semakin luas, volume pengolahan data semakin meningkat dan perubahan prinsip akuntansi yang baru.
2. Untuk meraih kesempatan-kesempatan (*opportunities*)  
Organisasi mulai merasakan bahwa teknologi informasi perlu digunakan untuk meningkatkan penyediaan informasi sehingga dapat mendukung dalam proses pengambilan keputusan yang akan dilakukan oleh manajemen.

3. Adanya instruksi-instruksi (*directives*)

Adanya instruksi-instruksi dari pimpinan ataupun dari luar organisasi, seperti peraturan pemerintah.

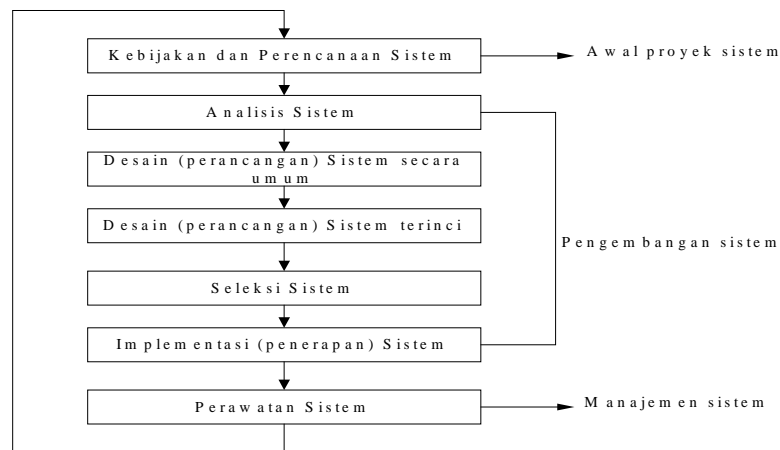
**2.2.2. Siklus Hidup Pengembangan Sistem**

Pengembangan sistem informasi yang berbasis komputer dapat merupakan tugas kompleks yang membutuhkan banyak sumber daya dan dapat memakan waktu berbulan-bulan bahkan bertahun-tahun untuk menyelesaikannya. Proses pengembangan sistem melewati beberapa tahapan dari mulai sistem itu direncanakan sampai dengan sistem itu diterapkan, dioperasikan dan dipelihara.

Bila operasi sistem yang telah dikembangkan masih timbul permasalahan-permasalahan yang kritis serta tidak dapat diatasi dalam tahap pemeliharaan sistem, maka perlu dikembangkan kembali suatu sistem untuk mengatasinya dan proses ini kembali ke tahap yang pertama, yaitu tahap perencanaan sistem. Siklus ini disebut dengan siklus hidup suatu sistem (*system life cycle*). (Jogiyanto,2005)

*System life cycle* adalah langkah-langkah (pedoman) yang harus diikuti untuk mengembangkan dan merancang sebuah sistem. Siklus hidup pengembangan sistem ini adalah seperti kompas di dalam merancang sistem.

Adapun langkah-langkah dari siklus hidup pengembangan sistem dapat dilihat pada gambar 2.2.



Gambar 2.2 Siklus Hidup Pengembangan Sistem

Berdasarkan gambar 2.2. dapat dijelaskan sebagai berikut :

### 1. Tahap Kebijakan Dan Perencanaan Sistem

Merupakan tahap awal dalam siklus pengembangan sistem, sebelum suatu sistem informasi dikembangkan, umumnya terlebih dahulu dimulai dengan adanya suatu analisis, kebijakan dan perencanaan untuk mengembangkan sistem itu. Tanpa adanya perencanaan sistem yang baik, pengembangan sistem tidak akan dapat berjalan sesuai dengan yang diharapkan. Perencanaan sistem (*system planning*) merupakan pedoman untuk melakukan pengembangan sistem.

Proses dari perencanaan sistem dikelompokkan dalam 3 proses utama, yaitu:

- 1) Merencanakan proyek-proyek sistem yang dilakukan oleh staf perencana sistem.
- 2) Menentukan proyek-proyek sistem yang akan dikembangkan yang dilakukan oleh komite pengarah.
- 3) Mendefinisikan proyek-proyek sistem dikembangkan yang dilakukan oleh analis sistem.

### 2. Tahap Analisis Sistem

Analisis sistem adalah penguraian dari suatu sistem informasi yang utuh kedalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya.

Analisis sistem dilakukan setelah tahap perencanaan sistem dan sebelum tahap disain sistem. Pada tahap ini diberikan penilaian terhadap sistem yang lama dan dipelajari kelemahan yang ada serta mencari beberapa alternatif pemecahan masalah. Di dalam tahap ini terdapat langkah-langkah dasar yang harus dilakukan yaitu:

- a. *Identify*, yaitu mengidentifikasi masalah, meliputi :
  - a) Mengidentifikasi penyebab masalah
  - b) Mengidentifikasi titik keputusan
  - c) Mengidentifikasi personil-personil kunci
- b. *Understand*, yaitu memahami kerja dari sistem yang ada, meliputi :
  - a) Menentukan jenis penulisan.
  - b) Merencanakan jadwal penulisan
  - c) Membuat penugasan penulisan.
  - d) Membuat agenda wawancara.
  - e) Mengumpulkan hasil penulisan.
- c. *Analyse*, yaitu menganalisis sistem.
  - a) Menganalisis kelemahan sistem
  - b) Menganalisis kebutuhan informasi pemakai atau manajemen
- d. *Raport*, yaitu membuat laporan hasil analisis.

Setelah proses analisis sistem ini selesai dilakukan, tugas berikutnya dari analisis sistem dan timnya adalah membuat laporan hasil analisis. Laporan ini akan diberikan ke steering committee yang nanti diteruskan ke pihak manajemen.

### 3. Tahap Perancangan Sistem Secara Umum

Setelah tahap analisis dilakukan, maka dilakukan pengembangan sistem secara umum dan menjelaskan informasi yang dihasilkan sistem tersebut. Tujuan tahap ini adalah untuk memberikan gambaran umum kepada user tentang sistem yang baru. Desain sistem secara umum mengidentifikasi komponen-komponen sistem informasi yang akan didesain secara rinci.

### 4. Tahap Perancangan Sistem Secara Terinci

Tahap ini merupakan perincian dari sistem secara umum, desain sistem secara umum ditransformasikan ke dalam bentuk yang lebih spesifik untuk membangun sebuah sistem. Desain terinci dimaksudkan untuk pemrogram komputer dan ahli teknik lainnya yang akan mengimplementasi sistem. Tujuan dari tahap ini adalah mendesain sistem baru yang dapat menyelesaikan masalah-masalah yang dihadapi organisasi secara logis.

#### 5. Tahap Seleksi Sistem

Tahap ini mencari beberapa penyebab permasalahan pada sistem lama dan memilih satu pemecahan masalah dari beberapa alternatif yang ada.

#### 6. Tahap Implementasi Sistem

Tahap ini merupakan tahap dimana suatu sistem siap untuk dioperasikan. Tahap ini terdiri dari langkah-langkah sebagai berikut:

- 1) Menerapkan rencana implementasi.
- 2) Melakukan kegiatan implementasi.
- 3) Tindak lanjut implementasi.

#### 7. Tahap Perawatan Sistem

Tahap ini adalah merupakan tahap akhir dalam sebuah pengembangan sistem. Dimana pada tahap ini hanya lebih ditekankan kepada pemeliharaan sistem, perawatan database dan menjaga kerja sistem yang sudah direncanakan di tahap pertama.

### **2.3 Pendekatan Berorientasi Objek**

Pendekatan Berorientasi Objek adalah cara memandang persoalan menggunakan model-model yang diorganisasikan seputar konsep objek yang mengkombinasikan struktur data dan perilaku suatu entitas. Pada pendekatan ini, organisasi perangkat lunak adalah sebagai kumpulan objek diskrit yang saling bekerja sama, berkomunikasi, dan berinteraksi menuju sasaran tertentu (Bambang Hariyanto, 2004).

#### **2.3.1 Object Oriented Analysis and Design (OOAD)**

OOAD adalah metode analisis yang memeriksa requirements dari sudut pandang kelas dan objek yang ditemui dalam ruang lingkup permasalahan yang mengarahkan arsitektur software yang didasarkan pada manipulasi objek-objek sistem atau subsistem. OOAD merupakan cara baru dalam memikirkan suatu masalah dengan menggunakan model yang dibuat menurut konsep sekitar dunia nyata. Dasar pembuatan adalah objek, yang merupakan kombinasi antara struktur data dan perilaku dalam satu entitas.

### 2.3.2 Metodologi dalam OOAD

Metodologi adalah cara sistematis untuk mengerjakan analisis dan disen. Dengan metodologi, pihak yang membangun system software dapat merencanakan dan mengulangi pekerjaan dilain waktu. Metodologi juga menghilangkan perbedaan notasi untuk suatu hal yang sama karena setiap orang akan berbicara dalam bahasa yang sama. Metodologi yang paling banyak dalam OOAD, yaitu : *Object Modeling Technique* (OMT) dari Rumbaugh, *Object Oriented Booch, Responsibility-Driven Design/ Class Responsibility Collaboration* (RDD/CRC) dari Wirf-Broock, *Metodologi Coad/ Yourdan* dan *Jacobson Object Oriented Software Engineering* (OOSE).

### 2.3.3 Konsep OOAD

OOAD mencakup analisis dan desain sebuah sistem dengan pendekatan objek, yaitu analisis berorientasi objek (OOA) dan desain berorientasi objek (OOD). OOA adalah metode analisis yang memeriksa requirement (syarat/keperluan) yang harus dipenuhi sebuah sistem) dari sudut pandang kelas-kelas dan objek-objek yang ditemui dalam ruang lingkup perusahaan. Sedangkan OOD adalah metode untuk mengarahkan arsitektur software yang didasarkan pada manipulasi objek-objek sistem atau subsistem.

Terdapat beberapa konsep dalam OOAD, yaitu :

1. Objek (*object*)
  - 1) Objek adalah benda secara fisik dan konseptual yang ada di sekitar kita. Sebuah objek memiliki keadaan sesaat yang disebut state.
  - 2) *State* dari sebuah objek adalah kondisi dari objek atau himpunan keadaan yang menggambarkan objek tersebut. State dinyatakan dengan nilai dari atribut objeknya.
  - 3) *Atribut* adalah nilai internal suatu objek yang mencerminkan karakteristik objek, kondisi sesaat, koneksi dengan objek lain dan identitas.
  - 4) *Behaviour* (perilaku objek) mendefinisikan bagaimana sebuah objek bertindak dan memberi reaksi. Behaviour ditentukan oleh himpunan



semua atau beberapa operasi yang dapat dilakukan oleh objek tersebut, yang dicerminkan oleh interface, service, dan method dari objek tersebut.

- 5) *Interface* adalah pintu untuk mengakses service dari objek
- 6) *Service* adalah fungsi yang dapat dikerjakan oleh sebuah objek
- 7) *Method* adalah mekanisme internal objek yang mencerminkan perilaku objek tersebut

## 2. Kelas (*class*)

*Class* adalah himpunan objek yang sejenis yaitu mempunyai sifat (*atribut*), perilaku umum (operasi), relasi umum dengan objek lain dan semantik umum. *Class* adalah abstraksi dari objek dalam dunia nyata. *Class* menetapkan spesifikasi perilaku dan atribut dari objek tersebut.

## 3. Kotak Hitam (*black boxes*)

Sebuah objek adalah kotak hitam. Konsep ini menjadi dasar implementasi objek. Dalam operasi OOAD hanya developer yang dapat memahami detail proses yang ada didalam kotak tersebut, sedangkan user tidak perlu mengetahui apa yang dilakukan yang penting mereka dapat menggunakan objek untuk memproses kebutuhan mereka. Kotak hitam berisi kode dan data.

- 1) *Encapsulation*, yaitu proses menyembunyikan detail implementasi sebuah objek. Untuk mengakses data objek tersebut adalah melalui interface. Untuk berkomunikasi dengan objek digunakan *message*.
- 2) *Message* adalah permintaan agar objek menerima untuk membawa metode yang ditunjukkan oleh perilaku dan mengembalikan result dari aksi tersebut kepada objek pengirim (*sender*)

## 4. Asosiasi dan Agregasi

- 1) Asosiasi adalah hubungan yang mempunyai makna antara sejumlah objek. Asosiasi digambarkan dengan sebuah garis penghubung diantara objeknya. Contohnya : Asosiasi karyawan dengan unit kerja. Setiap karyawan bekerja di satu unit kerja, sedangkan unit kerja dapat memiliki beberapa karyawan.
- 2) Agregasi adalah bentuk khusus sebuah asosiasi yang menggambarkan seluruh bagian pada suatu objek merupakan bagian dari objek yang lain.

Contohnya : Kopling dan piston adalah bagian dari mesin, sedangkan mesin, roda, body merupakan bagian dari sebuah mobil.

#### **2.3.4 OOA (*Object Oriented Analysis*)**

OOA mempelajari permasalahan dengan menspesifikasikannya atau mengobservasi permasalahan tersebut dengan menggunakan metode berorientasi objek. Biasanya analisa sistem dimulai dengan adanya dokumen permintaan (*requirement*) yang diperoleh dari semua pihak yang berkepentingan. (Mis: klien, developer, pakar, dll). Dokumen permintaan memiliki 2 fungsi yaitu : memformulasikan kebutuhan klien dan membuat suatu daftar tugas.

Analisis berorientasi obyek (OOA) melihat pada domain masalah, dengan tujuan untuk memproduksi sebuah model konseptual informasi yang ada di daerah yang sedang dianalisis. Model analisis tidak mempertimbangkan kendala-kendala pelaksanaan apapun yang mungkin ada, seperti konkurensi, distribusi, ketekunan, atau bagaimana sistem harus dibangun. Kendala pelaksanaan ditangani selama desain berorientasi objek (OOD). Analisis dilakukan sebelum *design*.

Sumber-sumber untuk analisis dapat persyaratan tertulis pernyataan, dokumen visi yang formal, wawancara dengan *stakeholder* atau pihak yang berkepentingan lainnya. Sebuah sistem dapat dibagi menjadi beberapa domain, yang mewakili bisnis yang berbeda, teknologi, atau bidang yang diminati, masing-masing dianalisis secara terpisah.

Hasil analisis berorientasi objek adalah deskripsi dari apa sistem secara fungsional diperlukan untuk melakukan, dalam bentuk sebuah model konseptual. itu biasanya akan disajikan sebagai seperangkat menggunakan kasus, satu atau lebih UML diagram kelas, dan sejumlah diagram interaksi. Tujuan dari analisis berorientasi objek adalah untuk mengembangkan model yang menggambarkan perangkat lunak komputer karena bekerja untuk memenuhi seperangkat persyaratan yang ditentukan pelanggan.

### **2.3.5 OOD (*Object Oriented Design*)**

OOD mengubah model konseptual yang dihasilkan dalam analisis berorientasi objek memperhitungkan kendala yang dipaksakan oleh arsitektur yang dipilih dan setiap non fungsional teknologi atau lingkungan kendala, seperti transaksi throughput, *response time*, *run* waktu platform, lingkungan pengembangan, atau bahasa pemrograman.

### **2.3.6 OOP (*Object Oriented Programming*)**

Pemrograman berorientasi Objek adalah sebuah teknik pemrograman yang dalam proses pengembangannya menggunakan terminologi objek. Di mana setiap objek memiliki atribut beserta dengan fungsi yang dapat saling berinteraksi satu dengan yang lain. (Sakur, 2010).

Seperti halnya objek jika kita memperhatikan sebuah objek makhluk hidup baik manusia ataupun binatang maka dapat mengenalinya dari bentuk, ukuran, beratnya kemudian dari prilakunya kita dapat melihat bahwa makhluk hidup dapat melihat, makan, berjalan, berlari, dan selain itu terdapat fungsi lainnya seperti fungsi peredaran darah yang berhubungan dengan fungsi pernapasan atau atau fungsi pencernaan.

### **2.3.7 Teknik Pemodelan dalam OOAD**

Model Objek :

1. Model objek Menggambarkan struktur statis dari suatu objek dalam sistem dan relasinya
2. Model objek berisi diagram objek. Diagram objek adalah graph dimana nodenya adalah kelas yang mempunyai relasi antar kelas.

Model Dinamik

1. Model dinamik menggambarkan aspek dari sistem yang berubah setiap saat.
2. Model dinamik dipergunakan untuk menyatakan aspek kontrol dari sistem.

3. Model dinamik berisi state diagram. State diagram adalah graph dimana nodenya adalah state dan arc adalah transisi antara state yang disebabkan oleh event.

#### Model Fungsional

- 1) Model fungsional menggambarkan transformasi nilai data di dalam sistem.
- 2) Model fungsional berisi data flow diagram. DFD adalah suatu *graph* dimana *nodenya* menyatakan proses dan *arcnya* adalah aliran data.

### 2.3.8 Karakteristik dari Objek

#### A. Objek

- a) Identitas berarti bahwa data diukur mempunyai nilai tertentu yang membedakan entitas disebut Objek.
- b) Objek dapat kongkrit, seperti halnya arsip dalam sistem, atau konseptual seperti kebijakan penjadwalan dalam multiprocessing pada sistem operasi.
- c) Setiap objek mempunyai sifat yang melekat pada identitasnya.
- d) Dua objek dapat berbeda walaupun bila semua nilai atributnya identik

#### B. Kelas Objek

- a) Kelas merupakan gambaran sekumpulan Objek yang terbagi dalam atribut, operasi, metode, hubungan, dan makna yang sama.
- b) Suatu kegiatan mengumpulkan data (atribut) dan perilaku (operasi) yang mempunyai struktur data sama ke dalam satu grup.
- c) Kelas Objek merupakan wadah bagi Objek. Dapat digunakan untuk menciptakan Objek.
- d) Objek mewakili fakta/keterangan dari sebuah kelas.

#### C. Istilah-istilah Objek

- a) Atribut : Data item yang menegaskan Objek
- b) Operasi : Fungsi di dalam kelas yang dikombinasikan ke bentuk tingkah laku kelas

- c) Metode : Pelaksanaan prosedur (badan dari kode yang mengeksekusi respon terhadap permintaan objek lain di dalam sistem).

### 2.3.9 Karakteritik Metodologi Berorientasi Objek

Metodologi pengembangan sistem berorientasi objek mempunyai tiga karakteristik utama :

#### A. *Encapsulation* (Pengkapsulan)

- a) *Encapsulation* merupakan dasar untuk pembatasan ruang lingkup program terhadap data yang diproses.
- b) Data dan prosedur atau fungsi dikemas bersama-sama dalam suatu objek, sehingga prosedur atau fungsi lain dari luar tidak dapat mengaksesnya.
- c) Data terlindung dari prosedur atau objek lain, kecuali prosedur yang berada dalam objek itu sendiri.

#### B. *Inheritance* (Pewarisan)

- a) *Inheritance* adalah teknik yang menyatakan bahwa anak dari objek akan mewarisi data/atribut dan metode dari induknya langsung.
- b) Atribut dan metode dari objek induk diturunkan kepada anak objek, demikian seterusnya.
- c) *Inheritance* mempunyai arti bahwa atribut dan operasi yang dimiliki bersama di antara kelas yang mempunyai hubungan secara hirarki.
- d) Suatu kelas dapat ditentukan secara umum, kemudian ditentukan spesifik menjadi subkelas.
- e) Setiap subkelas mempunyai hubungan atau mewarisi semua sifat yang dimiliki oleh kelas induknya, dan ditambah dengan sifat unik yang dimilikinya.
- f) Kelas Objek dapat didefinisikan atribut dan service dari kelas Objek lainnya.
- g) *Inheritance* menggambarkan generalisasi sebuah kelas

C. *Polymorphism* (perbedaan bentuk)

- a) *Polimorfisme* yaitu konsep yang menyatakan bahwa sesuatu yang sama dapat mempunyai bentuk dan perilaku berbeda.
- b) *Polimorfisme* mempunyai arti bahwa operasi yang sama mungkin mempunyai perbedaan dalam kelas yang berbeda.
- c) Kemampuan objek-objek yang berbeda untuk melakukan metode yang pantas dalam merespon message yang sama.
- d) Seleksi dari metode yang sesuai bergantung pada kelas yang seharusnya menciptakan Objek.

## 2.4 Model Umum Perancangan Analisis dan Perancangan Sistem

Adapun model perancangan analisis dan perancangan sistem ini adalah:

### 2.4.1 *Rational Rose*

Rational Rose merupakan salah satu tool yang digunakan membangun model suatu sistem secara visual yang memiliki banyak kemampuan untuk pembentukan sistem berorientasi obyek yang menggunakan UML (<http://latitude.east.asu.edu/415/projects/rose/rose-tutorial.pdf>).

Dalam UML terdapat beberapa istilah yang sering digunakan seperti : views, diagram dan elemen model.

1. *View*

Rational Rose memiliki empat view yaitu : *Use Case View*, *Logical View*, *Componen View* dan *Deployment View*.

2. *Diagram Rational Rose* memiliki delapan diagram yaitu : *Use case diagram*, *Sequence diagram*, *Collaboration diagram*, *Activity diagram*, *Class diagram*, *State diagram*, *Component diagram* dan *Deployment diagram*.

3. *Elemen Model*

Konsep-konsep yang digunakan dalam diagram merupakan elemen-elemen model yang menyatakan konsep berorientasi obyek secara

umum, seperti *class*, *object* dan *message*, serta hubungan antar konsep termasuk *association*, *dependency* dan *generalization*.

#### **2.4.2 Unified Modelling Language (UML)**

*UML* adalah sebuah bahasa yang berdasarkan grafik/gambar untuk memvisualisasi, menspesifikasikan, membangun, dan pendokumentasian dari sebuah sistem pengembangan software berbasis OO (*Object-Oriented*). *UML* juga merupakan sistem notasi yang membantu pemodelan sistem menggunakan konsep berorientasi objek.

##### **2.4.2.1 Definisi Umum UML (*Unified Modelling Language*)**

*UML (Unified Modelling Language)* adalah sebuah alat Bantu yang sangat handal di dunia pengembangan sistem berorientasi objek. Hal ini disebabkan karena *UML* menyediakan bahasa pemodelan visual yang memungkinkan bagi pengembangan sistem untuk membuat cetak biru atas visi mereka dalam bentuk baku, mudah dimengerti serta dilengkapi dengan mekanisme yang efektif untuk berbagi (*sharing*) dan mengkomunikasikan rancangan mereka dengan yang lain. (Adi Nugroho, 2005)

*Unified Model Language (UML)* adalah bahasa universal untuk memvisualisasikan grafis model yang tepat, menetapkan model yang tepat, lengkap, dan tidak ambigu untuk mengambil semua keputusan penting dalam analisis, desain dan implementasi, membangun model yang dapat dihubungkan langsung dengan bahasa pemrograman, mendokumentasikan semua informasi yang dikumpulkan oleh tim sehingga memungkinkan untuk berbagi informasi

*UML* dibangun atas model 4+1 *view*. Model ini didasarkan pada fakta struktur sebuah sistem dideskripsikan dalam 5 *view*, dimana salah satunya adalah *Use Case View*.

Dalam proyek pengembangan sistem apapun, fokus utama dalam analisis dan perancangan adalah model. Hal ini berlaku umum tidak hanya untuk perangkat lunak. Dengan model kita dapat merepresentasikan sesuatu karena : (Adi Nugroho, 2005).

1. Model mudah dan cepat untuk dibuat.
2. Model bisa digunakan sebagai simulasi untuk mempelajari lebih detail tentang sesuatu.
3. Model bisa dikembangkan sejalan dengan pemahaman kita tentang sesuatu..
4. Model bisa mewakili sesuatu yang nyata maupun tidak nyata.

#### **2.4.2.2 Diagram-Diagram *UML (Unified Modelling Language)***

*UML* mempunyai sejumlah elemen grafis yang bisa dikombinasikan menjadi diagram. Karena ini merupakan sebuah bahasa, *UML* mempunyai aturan untuk menggabungkan dan mengkombinasikan elemen-elemen tersebut.

Dalam membangun suatu model perangkat lunak dengan *UML*, digunakan bentuk-bentuk diagram atau simbol untuk merepresentasikan elemen-elemen dalam sistem. Bentuk diagram yang digunakan untuk merepresentasikannya adalah sebagai berikut : sumber (Adi Nograho, 2005)

1. *Use case Diagram*
2. *Activity Diagram*
3. *Sequence diagram*
4. *Class Diagram*
5. *Collaboration Diagram*
6. *State Diagram*
7. *Component Diagram*
8. *Deployment Diagram*



Tabel 2.1 Tipe Diagram *UML*



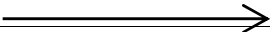
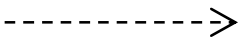
<b>Diagram</b>	<b>Tujuan</b>
<i>Use Case</i>	Menunjukkan sekumpulan kasus fungsional dan aktor dan hubungannya.
<i>Activity</i>	Pandangan operasi, bagaimana objek-objek bekerja, aksi-aksi yang mempengaruhi obyek, pandangan <i>use case workflow</i> .
<i>Sequence</i>	Berfungsi untuk <i>overview</i> perilaku sistem, menunjukkan objek-objek yang diperlukan, mendokumentasikan skenario dari suatu diagram <i>Use Case</i> , memeriksa jalur-jalur pengaksesan.
<i>Class</i>	Memodelkan kosakata di sistem, distribusi dan tanggung jawab, tipe primitif, kolaborasi, skema <i>database</i> logik.
<i>Collaboration</i>	Memodelkan pandangan perilaku sistem pada <i>link-link</i> di antara objek-objek. Ilustrasi dari <i>use case</i> , memeriksa jalur-jalur pengaksesan
<i>Statechart</i>	Pandangan objek secara waktu, pandangan dalam berkaitan dengan ransangan eksternal.
<i>Component</i>	Memodelkan <i>file</i> yang dapat dieksekusi dan pustaka, memodelkan tabel, <i>file</i> dan dokumen, memodelkan API ( <i>Application Programming Interupt</i> )
<i>Deployment</i>	Konfigurasi pemrosesan saat jalan dan komponen-komponen yang terdapat didalamnya.

### 2.4.2.3 Diagram-Diagram UML Yang Digunakan

#### 1. Use case Diagram

Diagram *Use case* merupakan salah satu diagram untuk memodelkan aspek perilaku sistem. Masing-masing diagram *use case* menunjukkan sekumpulan *use-case*, aktor, dan hubungannya. Diagram *use case* adalah penting untuk memvisualisasikan, memspesifikasikan, dan mendokumentasikan kebutuhan perilaku sistem. Diagram *use-case* merupakan pusat pemodelan perilaku sistem, subsistem, kelas. Berikut adalah elemen dalam *use case* :

Tabel 2.2 Notasi Use Case Diagram

Penjelasan	Notasi UML
Aktor : Mewakili peran orang, sistem yang lain atau alat ketika berkomunikasi dengan <i>use case</i> .	
<i>Use Case</i> : Abstraksi dari interaksi antara sistem dan <i>actor</i>	
<i>Association</i> : adalah abstraksi dari penghubung antara <i>actor</i> dan <i>use case</i>	
Generalisasi : menunjukkan spesialisasi actor untuk dapat berpartisipasi dalam <i>use case</i>	

#### 2. Activity Diagram

*Activity diagrams* menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi.

*Activity diagram* merupakan *state diagram* khusus, di mana sebagian besar *state* adalah *action* dan sebagian besar transisi di-*trigger* oleh selesainya *state* sebelumnya (*internal processing*). Oleh karena itu *activity diagram* tidak

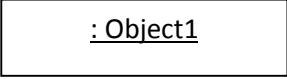



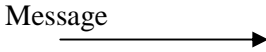
menggambarkan behaviour internal sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum.

Sebuah aktivitas dapat direalisasikan oleh satu *use case* atau lebih. Aktivitas menggambarkan proses yang berjalan, sementara *use case* menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas. Sama seperti *state*, standar UML menggunakan segiempat dengan sudut membulat untuk menggambarkan aktivitas. *Decision* digunakan untuk menggambarkan behaviour pada kondisi tertentu. Untuk mengilustrasikan proses-proses paralel (*fork* dan *join*) digunakan titik sinkronisasi yang dapat berupa titik, garis horizontal atau vertikal. *Activity diagram* dapat dibagi menjadi beberapa *object swimlane* untuk menggambarkan objek mana yang bertanggung jawab untuk aktivitas tertentu.

### 3. *Sequence diagram*

*Sequence diagram* mendokumentasikan komunikasi/interaksi antar kelas-kelas. Diagram ini menunjukkan sejumlah obyek dan *message* (pesan) – yang diletakkan diantara obyek-obyek didalam *use case*. Perlu diingat bahwa di dalam diagram ini, kelas-kelas dan aktoraktor diletakkan dibagian atas diagram dengan urutan dari kiri ke kanan dengan garis *lifeline* yang diletakkan secara vertikal terhadap kelas dan aktor. Berikut adalah notasi-notasinya :

Tabel 2.3 Notasi *Sequence Diagram*

<i>Nama</i>	<i>Keterangan</i>	Notasi
<i>Object</i>	<i>Object</i> merupakan instance dari sebuah class dan dituliskan tersusun secara horizontal. Digambarkan sebagai sebuah class (kotak) dengan nama obyek didalamnya yang diawali dengan sebuah titik koma	
<i>Actor</i>	<i>Actor</i> juga dapat berkomunikasi dengan object, maka actor juga dapat diurutkan sebagai kolom. Simbol Actor sama dengan simbol pada Actor Use Case Diagram.	
<i>Lifeline</i>	<i>Lifeline</i> mengindikasikan keberadaan sebuah object dalam basis waktu. Notasi untuk Lifeline adalah garis putus-putus vertikal yang ditarik dari sebuah obyek.	
<i>Activation</i>	<i>Activation</i> dinotasikan sebagai sebuah kotak segi empat yang digambar pada sebuahlifeline.  Activation mengindikasikan sebuah obyek yang akan melakukan sebuah aksi.	
<i>Message</i>	<i>Message</i> , digambarkan dengan anak panah horizontal antara <i>Activation</i> . <i>Message</i> mengindikasikan komunikasi antara <i>object-object</i> .	

#### 4. *Class Diagram*

*Class* adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda/fungsi).

*Class diagram* menggambarkan struktur dan deskripsi *class*, *package* dan objek beserta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi, dan lain-lain. *Class* memiliki tiga area pokok :

1. Nama (dan stereotype)
2. Atribut
3. Metoda, Atribut dan metoda dapat memiliki salah satu sifat berikut :
  - a) *Private*, tidak dapat dipanggil dari luar *class* yang bersangkutan
  - b) *Protected*, hanya dapat dipanggil oleh *class* yang bersangkutan dan anak-anak yang mewarisinya
  - c) *Public*, dapat dipanggil oleh siapa saja

*Class* dapat merupakan implementasi dari sebuah *interface*, yaitu *class* abstrak yang hanya memiliki metoda. *Interface* tidak dapat langsung diinstansiasikan, tetapi harus diimplementasikan dahulu menjadi sebuah *class*. Dengan demikian *interface* mendukung resolusi metoda pada saat *run-time*.

#### 5. *Collaboration Diagram*


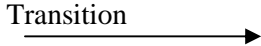


*Collaboration diagram* menggunakan prinsip yang sama dengan *sequence diagram*, sama-sama memodelkan interaksi antar obyek-obyek, yang membedakannya hanya cara penggambarannya saja. Pada *collaboration diagram* ini, obyek-obyek dan *message* (pesan) yang ada digambarkan mirip seperti flowchart, hanya saja, untuk menjaga urutan pesan yang diterima oleh masing-masing obyek, pesan-pesan tersebut diberi nomor urutan pesan.

#### 6. *State Diagram*

*Statechart diagram*, atau yang biasa juga disebut *state diagram* digunakan untuk mendokumentasikan beragam kondisi/keadaan yang bisa terjadi terhadap sebuah *class* dan kegiatan apa saja yang dapat merubah kondisi/keadaan tersebut. Contohnya sebuah televisi yang dapat berada dalam kondisi menyala atau mati,

jika tombol “power” ditekan maka televisi akan menyala, begitu juga sebaliknya akan mati jika tombol “power” ditekan kembali. Maka disini kita mempunyai sebuah kelas yaitu televisi, 2 *state* yaitu *menyala* dan *mati* dan 2 *transition* yaitu *menyalakan TV* dan *mematikan TV*. Tidak seperti diagram-diagram *behavioural* lainnya yang memodelkan interaksi diantara beberapa *class*, *state diagram* justru biasanya hanya memodelkan transisi yang terjadi hanya pada sebuah *class*. Berikut adalah notasi *state diagram* :

Tabel 2.4 Notasi Statechart Diagram

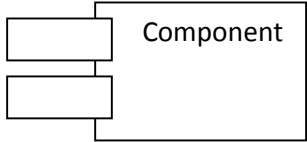

<i>Nama</i>	Keterangan	<i>Notasi</i>
<i>State</i>	Notasi <i>State</i> menggambarkan kondisi sebuah entitas, dan digambarkan dengan segiempat yang pinggirnya tumpul dengan nama state didalamnya.	
<i>Transition</i>	Sebuah <i>Transition</i> menggambarkan sebuah perubahan kondisi objek yang disebabkan oleh sebuah event. Transition digambarkan dengan sebuah anak panah dengan nama event yang ditulis di atasnya, dibawahnya atau sepanjang anak panah tersebut.	
<i>Initial state</i>	<i>Initial State</i> adalah sebuah kondisi awal sebuah object sebelum ada perubahan keadaan. Initial State digambarkan dengan sebuah lingkaran solid. Hanya satu Initial State yang diizinkan dalam sebuah diagram	
<i>Final State</i>	<i>Final State</i> menggambarkan ketika objek berhenti memberi respon terhadap sebuah event. Final State digambarkan dengan lingkaran solid didalam sebuah lingkaran kosong.	

## 7. Component Diagram

Komponen perangkat lunak adalah bagian fisik dari sebuah sistem yang menetap di komputer. komponen merupakan implementasi software dari sebuah class. Komponen bias berupa tabel, *file data*, *file exe*, file DLL, dokumen dan lain-lain.

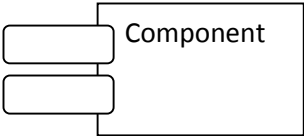
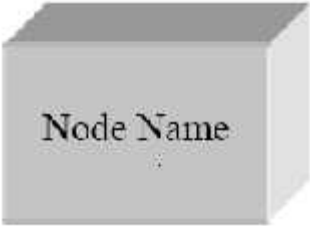

*Component diagram* mengandung komponen, *interface* dan *relationship*. Komponen diagram ini digunakan pada saat anda ingin memecah sistem menjadi komponen-komponen dan ingin menampilkan hubungan-hubungan mereka dengan antarmuka atau pemecahan komponen menjadi struktur yang lebih rendah. Secara umum dapat kita katakan bahwa component diagram kita gunakan untuk menjelaskan kebergantungan antar beragam komponen-komponen *software* seperti misalnya kebergantungan antara *file-file executable* dengan *file-file sumbernya (source file)* dan lain-lain. Berikut adalah notasi dari *component diagram*:

Tabel 2.5 Notasi Component Diagram

<i>Nama</i>	Keterangan	Notasi
<i>Component</i>	Sebuah komponen melambangkan sebuah entitas software dalam sebuah sistem. Sebuah komponen dinotasikan sebagai sebuah kotak segiempat dengan dua kotak kecil tambahan yang menempel disebelah kirinya.	
<i>Depedency</i>	Sebuah Dependency digunakan untuk menotasikan relasi antara dua komponen. Notasinya adalah tanda panah putus-putus yang diarahkan kepada komponen tempat sebuah komponen itu bergantung.	

Deployment diagram menunjukkan tata letak sebuah sistem secara fisik, menampakkan bagian-bagian *software* yang berjalan pada bagian-bagian *hardware* yang digunakan untuk mengimplementasikan sebuah sistem dan keterhubungan antara komponen-komponen *hardware* tersebut. *Deployment diagram* dapat digunakan pada bagian-bagian awal proses perancangan sistem untuk mendokumentasikan arsitektur fisik sebuah sistem. Berikut adalah notasi-notasi yang digunakan pada *deployment diagram* :

Tabel 2.6 Notasi Deployment Diagram

<i>Nama</i>	Keterangan	<i>Notasi</i>
<i>Component</i>	Pada deployment diagram, komponen-komponen yang ada diletakkan didalam node untuk memastikan keberadaan posisi mereka.	
<i>Node</i>	Node menggambarkan bagian-bagian hardware dalam sebuah sistem. Notasi untuk node digambarkan sebagai sebuah kubus 3 dimensi.	
<i>Association</i>	Sebuah association digambarkan sebagai sebuah garis yang menghubungkan dua node yang mengindikasikan jalur komunikasi antara element-element hardware.	

## 2.5 Penjualan

Dalam suatu perusahaan penjualan merupakan hal yang harus di perhatikan, karena hal itu merupakan sumber kehidup bagi suatu perusahaan.



### **2.5.1 Definisi Penjualan**

Penjualan adalah suatu proses kegiatan yang dipengaruhi oleh berbagai faktor social, budaya, politik dan ekonomi dan manajerial. Akibat dari berbagai faktor tersebut masing-masing individu maupun kelompok mendapatkan kebutuhan dan keinginan dengan menciptakan, menawarkan, dan menukarkan produk yang memiliki nilai komuditas. (Purnama, 2010).

Menurut Joel G. Siegel dan Joe K. Shim yang diterjemahkan oleh Moh. Kurdi, Penjualan adalah penerimaan yang diperoleh dari pengiriman barang dagangan atau dari penyerahan pelayanan dalam bursa sebagai barang pertimbangan. Pertimbangan ini dapat dalam bentuk tunai peralatan kas atau harta lainnya. Pendapatan dapat diperoleh pada saat penjualan, karena terjadi pertukaran, harga jual dapat ditetapkan dan bebannya diketahui. (Purnama, 2010).

Dari beberapa definisi diatas, Nampak dengan jelas bahwa penjualan itu merupakan sejumlah kegiatan yang terjadi akibat perpindahan barang atau jasa dari tangan produsen hingga ke tangan konsumen dan tercakup di dalamnya segala kegiatan sebelum dan sesudah pemindahan barang atau jasa tersebut.

### **2.5.2 Sistem Penjualan**

Sistem penjualan dibagi menjadi dua yaitu:

1. Sistem penjualan tunai

Merupakan transaksi yang apabila barang dan jasa diserahkan ke pembeli setelah perusahaan menerima uang dari pembeli.

2. Sistem penjualan kredit

Menurut Soemarso SR, “ penjualan kredit adalah penjualan barang dagang secara tidak tunai yang dicatat sebagai debit pada perkiraan piutang dagang dan kredit pada perkiraan penjualan.

### **2.5.3 Sistem Informasi Penjualan**

Sistem informasi penjualan diartikan sebagai suatu pembuatan pernyataan penjualan, kegiatan akan dijelaskan melalui prosedur-prosedur yang meliputi urutan kegiatan sejak diterimanya pesanan dari pembeli , pengecekan barang ada

atau tidak ada dan diteruskan dengan pengiriman barang yang disertai dengan pembuatan faktur dan mengadakan pencatatan atas penjualan yang berlaku. (Rhido Purnama 2010).

Sistem Informasi Penjualan adalah suatu sistem informasi yang mengorganisasikan serangkaian prosedur dan metode yang dirancang untuk menghasilkan, menganalisa, menyebarkan dan memperoleh informasi guna mendukung pengambilan keputusan mengenai penjualan.

## 2.6 Konsep Dasar *E-Commerce*

Perkembangan teknologi saat ini telah merubah gaya kehidupan manusia, apalagi dengan adanya dunia internet. Gaya hidup manusia dari waktu ke waktu senantiasa menghendaki keadaan yang makin efisien dan efektif bahkan untuk urusan belanja sekalipun. Untuk urusan belanja ini, maka salah satu keunggulan baru dari internet yang ini banyak di gemari banyak orang adalah *e-commerce*. Fasilitas *e-commerce* ,menawarkan berbagai kemudahan dalam melakukan transaksi di internet. Toko-toko *online* bertambah dari waktu ke waktu dan mudah dijumpai dengan berselancar di internet. Pada akhirnya manusia semakin dimanjakan, tetapi bukanlah suatu mustahil di era internet. (Triton, PB, 2006. Dikutip dari skripsi Rio Saputra, 2006).

Istilah *e-commerce* dapat merujuk pada interaksi yang memungkinkan oleh teknologi antara individu dan organisasi. Kategori *e-commerce* terbagi menjadi:

Tabel 2.7 Tipe *Electronic Commerce*

Jenis <i>e-commerce</i>	Karakteristik
<i>B to C (business to consumers)</i>	<ol style="list-style-type: none"> <li>1. Antara organisasi dengan perorangan</li> <li>2. Nilai uang yang dilibatkan lebih kecil</li> <li>3. Transaksi sewaktu-waktu atau tidak sering</li> <li>4. Secara relatif sederhana</li> </ol>

<i>B to B (business to business)</i>	<ol style="list-style-type: none"> <li>1. Antar organisasi</li> <li>2. Nilai uang yang dilibatkan lebih besar</li> <li>3. Hubungan yang kuat dan berkelanjutan</li> <li>4. Pemberian kredit oleh penjual ke pelanggan</li> </ol>
--------------------------------------	--

### 2.6.1 Pengertian *E-Commerce*

Menurut Esprit yang dikutip oleh Whiteley (2000) *e-commerce* adalah konsep umum setiap bentuk dari transaksi bisnis atau pertukaran informasi yang dilaksanakan dengan menggunakan teknologi komunikasi dan informasi. *E-commerce* digunakan antar perusahaan, antara perusahaan dengan pelanggan mereka, atau antara perusahaan dengan administrasi publik. *E-commerce* terdiri dari perdagangan produk, jasa dan barang-barang elektronik.

Pada dasarnya *e-commerce* adalah melakukan bisnis online. Dalam bentuknya yang paling jelas, *e-commerce* menjual produk kepada konsumen secara online, tapi faktanya jenis bisnis apa pun yang dilakukan secara elektronik adalah *e-commerce*. Sederhananya, *e-commerce* adalah membuat, mengelola, dan meluaskan hubungan komersial secara online (Kienan, 2000).

Usaha *e-commerce* yang paling baik dikembangkan bukan untuk menjual kepada konsumen online, melainkan untuk melayani orang atau lembaga lain yang telah ada dalam *e-commerce* maupun bisnis lapangan yang berharap untuk terlibat di sana (Kienan, 2000).

*Electronic Commerce* (Perniagaan Elektronik), sebagai bagian dari *Electronic Business* (bisnis yang dilakukan dengan menggunakan *electronic transmission*), oleh para ahli dan pelaku bisnis dicoba dirumuskan definisinya. Secara umum *e-commerce* dapat didefinisikan sebagai segala bentuk transaksi perdagangan/perniagaan barang atau jasa (*trade of goods and service*) dengan menggunakan media elektronik. Jelas, selain dari yang telah disebutkan di atas, bahwa kegiatan perniagaan tersebut merupakan bagian dari kegiatan bisnis. Kesimpulannya, "*e-commerce is a part of e-business*".

Media elektronik yang dibicarakan di dalam tulisan ini untuk sementara hanya difokuskan dalam hal penggunaan media internet. Pasalnya, penggunaan internetlah yang saat ini paling populer digunakan oleh banyak orang, selain merupakan hal yang bisa dikategorikan sebagai hal yang sedang ‘*booming*’. Perlu digaris bawahi, dengan adanya perkembangan teknologi di masa mendatang, terbuka kemungkinan adanya penggunaan media jaringan lain selain internet dalam *e-commerce*. Jadi pemikiran kita jangan hanya terpaku pada penggunaan media internet belaka.

Menurut Robert E. Johnson *e-commerce* merupakan suatu tindakan melakukan transaksi i-bisnis secara elektronik dengan menggunakan internet sebagai media komunikasi yang paling utama. Pada website ECARM (*The Society For Electronic Commerce And Rights Management*) dijelaskan bahwa *e-commerce* secara umum menunjukkan seluruh bentuk transaksi yang berhubungan dengan aktifitas-aktifitas perdagangan, termasuk organisasi dan perorangan yang berdasarkan pada pemrosesan dan transmisi data digital termasuk teks, suara, dan gambar-gambar visual (<http://www.cimcor.com>).

Menurut Coulter dan Buddemeir (2002) : *e-commerce* berhubungan dengan penjualan, periklanan, pemesanan produk, yang semuanya dikerjakan melalui internet. Beberapa perusahaan memilih untuk menggunakan kegiatan bisnis ini sebagai tambahan metode bisnis tradisional, sementara yang lainnya menggunakan internet secara eksklusif untuk mendapatkan para pelanggan yang berpotensi.

### **2.6.2 Konsep E-Commerce**

Bagi pihak konsumen, menggunakan *E-Commerce* dapat membuat waktu berbelanja menjadi singkat. Tidak ada lagi berlama-lama mengelilingi pusat pertokoan untuk mencari barang yang diinginkan. Selain itu, harga barang-barang yang dijual melalui *E-Commerce* biasanya lebih murah dibandingkan dengan harga di toko, karena jalur distribusi dari produsen barang ke pihak penjual lebih singkat dibandingkan dengan toko konvensional.

*Online shopping* menyediakan banyak kemudahan dan kelebihan jika dibandingkan dengan cara belanja yang konvensional. Selain bisa menjadi lebih cepat, di internet telah tersedia hampir semua macam barang yang biasanya dijual secara lengkap. Selain itu, biasanya informasi tentang barang jualan tersedia secara lengkap, sehingga walaupun kita tidak membeli secara *on-line*, kita bisa mendapatkan banyak informasi penting yang diperlukan untuk memilih suatu produk yang akan dibeli (Coulter dan Buddemeir, 2002).

### **2.6.3 Karakteristik E-Commerce**

Berbeda dengan transaksi perdagangan biasa, transaksi *e-commerce* memiliki beberapa karakteristik yang sangat khusus, yaitu :

a. Transaksi tanpa batas

Sebelum era internet, batas-batas geografi menjadi penghalang suatu perusahaan atau individu yang ingin go-international. Sehingga, hanya perusahaan atau individu dengan modal besar yang dapat memasarkan produknya ke luar negeri. Dewasa ini dengan internet pengusaha kecil dan menengah dapat memasarkan produknya secara internasional cukup dengan membuat situs web atau dengan memasang iklan di situs-situs internet tanpa batas waktu (24 jam), dan tentu saja pelanggan dari seluruh dunia dapat mengakses situs tersebut dan melakukan transaksi secara *online*.

b. Transaksi anonim

Para penjual dan pembeli dalam transaksi melalui internet tidak harus bertemu muka satu sama lainnya. Penjual tidak memerlukan nama dari pembeli sepanjang mengenai pembayarannya telah diotorisasi oleh penyedia sistem pembayaran yang ditentukan, yang biasanya dengan kartu kredit.

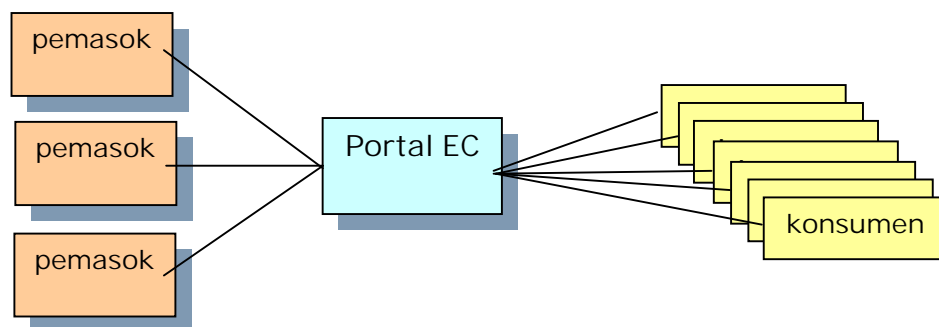
c. Produk digital dan non digital

Banyak perusahaan yang bergerak di bidang *e-commerce* dengan menawarkan barang tak berwujud seperti data, *software* dan ide-ide yang dijual melalui internet.

Implementasi *e-commerce* pada dunia industri yang penerapannya semakin lama semakin luas tidak hanya mengubah suasana kompetisi menjadi semakin dinamis dan global, namun telah membentuk suatu masyarakat tersendiri yang dinamakan *Komunitas Bisnis Elektronik (Electronic Business Community)*. Komunitas ini memanfaatkan *cyberspace* sebagai tempat bertemu, berkomunikasi, dan berkoordinasi ini secara intens memanfaatkan media dan infrastruktur telekomunikasi dan teknologi informasi dalam menjalankan kegiatannya sehari-hari (Rahardjo, 1999).

Seperti halnya pada masyarakat tradisional, pertemuan antara berbagai pihak dengan beragam kepentingan secara natural telah membentuk sebuah pasar tersendiri tempat bertemunya permintaan (*demand*) dan penawaran (*supply*). Transaksi yang terjadi antara *demand* dan *supply* dapat dengan mudah dilakukan walaupun yang bersangkutan berada dalam sisi geografis yang berbeda karena kemajuan dan perkembangan teknologi informasi, yang dalam hal ini adalah teknologi *e-commerce* (Purbo, 2006).

Secara umum *e-commerce* dapat diklasifikasikan menjadi dua jenis yaitu; *Business to Business (B2B)* dan *Business to Consumer (B2C)*. ***Business to Business (B2B)*** adalah sistem komunikasi bisnis on-line antar pelaku bisnis, sedangkan ***Business to Consumer (B2C)*** merupakan mekanisme toko on-line (*electronic shopping mall*), yaitu transaksi antara *e-merchant* dengan *e-customer* (Mardiyanto, 2008). Model *Business to Business (B2B)* dapat dilihat pada gambar berikut :



Gambar 2.3. Model *Business to Business (B2B)*. (Sumber: Stiawan, 2002).

Dalam *Business to Business* pada umumnya transaksi dilakukan oleh para *trading partners* yang sudah saling kenal dengan format data yang telah disepakati bersama. Sedangkan dalam *Business to Customer* sifatnya terbuka untuk publik, sehingga setiap individu dapat mengaksesnya melalui suatu web server :

- a. Servis yang diberikan bersifat umum (*generic*) dengan mekanisme yang dapat digunakan oleh khalayak ramai. Sebagai contoh, karena sistem Web sudah umum digunakan maka servis diberikan dengan menggunakan basis Web.
- b. Servis diberikan berdasarkan permohonan (*on demand*). Konsumer melakukan inisiatif dan produser harus siap memberikan respon sesuai dengan permohonan.

Pendekatan *client/server* sering digunakan dimana diambil asumsi client (*consumer*) menggunakan sistem yang minimal (berbasis Web) dan processing (*business procedure*) diletakkan di sisi server (Stiawan, 2002).

#### **2.6.4 Komponen-Komponen *E-Commerce***

Menurut Stiawan (2002) komponen-komponen *e-commerce* terdiri dari :

- a. Sistem Pembayaran Elektronik  
Sistem pembayaran elektronik diperlukan oleh sembarang bisnis yang menjual barang dan jasa secara online. Bisnis membutuhkan beberapa metode untuk: menerima pembayaran selama pelanggan sedang online, mengotentikasi pelanggan, dan melindungi privasi detail transaksi.
- b. Protokol  
TCP/IP menyediakan protokol paling banyak yang diperlukan untuk melengkapi transaksi bisnis secara online, tapi protokol khusus dibutuhkan untuk mengenkripsi informasi transaksi dan memelihara privasi pelanggan. Protokol yang sering digunakan adalah sebagai berikut:

1) SSL

Salah satu protokol yang umum digunakan adalah *Secure Socket Layer* (SSL). SSL menggunakan kunci, sertifikat/tanda tangan digital, dan enkripsi untuk melindungi informasi dan mengotentikasi pelanggan dan penjual.

2) SET

Protokol lain yang digunakan adalah *Secure Electronic Transactions* (SET), yang disponsori oleh VISA, Master Card, dan American Express. Perbedaan antara SET dan SSL adalah bahwa SET mengkonfirmasi ketersediaan dana selama dua pihak sedang online, dan melindungi informasi di antara ketiga pihak. SET juga menggunakan enkripsi yang kuat dan sertifikat/tanda tangan digital. SET menggunakan suatu kriptografi khusus yang dinamakan *asymmetric cryptography* untuk menjamin keamanan suatu transaksi. *Asymmetric cryptography* ini juga disebut dengan nama *Public-key Cryptography*. Enkripsi ini menggunakan dua kunci (berupa kode), satu kunci digunakan untuk meng-enkripsi data, dan kunci lainnya untuk men-dekripsi data tersebut. Kedua kunci tersebut terhubung secara matematis dengan rumus tertentu, sehingga data yang telah di-enkripsi oleh suatu kunci hanya bisa di-dekripsi dengan menggunakan kunci pasangannya.

## **2.6.5 Jenis-Jenis *E-Commerce***

Dilihat dari jenisnya e-commerce terbagi atas dua, yaitu:

### **2.6.5.1 *Business to Consumers (B2C)***

Penjual adalah suatu organisasi dan pembeli adalah individu. Aplikasi *B2C* adalah aplikasi yang digunakan untuk melakukan transaksi antara suatu organisasi dengan konsumen. (Sutarman, 2009).



Contoh-contoh dari B2C adalah : (Janner Simarmata, 2006)

1. *Advertising Revenue Model*

Merupakan suatu situs menawarkan informasi kepada para penggunanya atas jasa dan produk, dan menyediakan suatu kesempatan bagi penyedia untuk mengiklankannya. Perusahaan kemudian menerima pembayaran dari iklan tersebut. Seperti Yahoo.com memperoleh pendapatan utamanya dari penjualan iklan seperti banner ads.

2. *Subscription revenue model*

Merupakan suatu situs web menawarkan isi dan ongkos jasa atau biaya langganan untuk mengakses situs tersebut. Contohnya adalah Consumer Report Online <<http://www.consumerreports.org>>.

3. *Transaction Fee Model*

Suatu perusahaan menerima pembayaran untuk melakukan suatu transaksi. Sebagai contoh, Orbitz <<http://www.orbitz.com>> menawarkan harga pembayaran yang kecil pada konsumen ketika memesan tempat pada perusahaan penerbangan.

4. *Sales Revenue Model*

Merupakan perusahaan menjual barang-barang, informasi, atau jasa secara langsung ke pelanggan. <<http://www.amazon.com>> terutama menjual buku dan musik, <<http://www.travelocity.com>> suatu perusahaan penerbangan dan penyedia reservasi hotel.

#### **2.6.5.2 Business to Business (B2B)**

Pada B2B, penjual, pembeli dan transaksi terlibat dalam organisasi. Dengan B2B memungkinkan suatu perusahaan untuk membina suatu hubungan secara elektronik dengan distributor, penjual, supplier pelanggan, dan rekan bisnis. Terdapat beberapa model aplikasi B2B, yaitu : (Sutarman,2009).

1. *Sell-Side Marketplace*

Pada model ini, organisasi berusaha untuk menjual barang-barang atau layanan keorganisasi lain secara elektronik. Mekanisme kunci pada model

ini adalah katalog elektronik yang dapat dimodifikasi untuk setiap pembelian skala besar.

## 2. *Buy-Side Marketplace*

*Buy-side marketplace* juga dikenal dengan nama *e-procurement*, adalah model dimana teknologi *e-commerce* digunakan untuk menyederhanakan proses pembelian dengan tujuan untuk mengurangi biaya dari barang yang dibeli, biaya administrasi pengadaan barang, dan waktu siklus pembelian.

### 2.6.6 Keuntungan *E-Commerce*

Keuntungan *e-commerce* beraneka ragam dibagi berdasarkan pihak yang diuntungkan, seperti diuraikan sebagai berikut : (Sutarman, 2009).

#### A. Untuk organisasi

- a) Mempeluas pasar perusahaan ke pasar nasional, bahkan internasional.
- b) Memungkinkan perusahaan untuk memperoleh barang atau layanan dari perusahaan lain secara cepat dengan biaya yang minimal.
- c) Mempesingkat atau mengurangi jalur distribusi pasar (marketing distribution channel). Barang menjadi murah dan keuntungan menjadi lebih tinggi.
- d) Dapat mengurangi inventori barang dengan memfasilitasi *pull-type supply chain management*. Ini memungkinkan modifikasi produk dan mengurangi biaya inventori.
- e) Membantu bisnis kecil untuk bersaing dengan perusahaan-perusahaan besar.
- f) Mengurangi biaya pembuatan, proses, penyaluran, penyimpanan, dan mendapatkan informasi dengan adanya proses digital.

#### B. Untuk Pelanggan

- a) Menyediakan produk-produk dan layanan yang tidak mahal dengan memungkinkan konsumen untuk melakukan perbandingan secara *online*.
- b) Memberikan konsumen lebih banyak pilihan.

- c) Memungkinkan konsumen untuk melakukan transaksi dalam 24 jam sehari, dimana saja.
- d) Memberikan informasi yang relevan dalam hitungan detik.
- e) Memungkinkan konsumen untuk mendapatkan produk yang dapat dimodifikasi dengan harga yang bersaing.
- f) memungkinkan konsumen untuk berinteraksi pada komunitas elektronik dan untuk bertukar pikiran dan membandingkan pengalaman.

#### C. Bagi Masyarakat

- a) Memungkinkan tiap individu untuk bekerja di rumah dan mengurangi mobilitas sehingga mengurangi kepadatan jalan raya, polusi dan udara.
- b) Memungkinkan barang-barang dijual dengan harga yang rendah sehingga meningkatkan taraf hidup masyarakat.

### **2.6.7 Dampak Positif dan Negatif *E-Commerce*.**

Didalam dunia *E-Commerce* pasti terdapat dampak positif dan negatifnya.

Dampak positifnya, yaitu :

- a) Revenue Stream (aliran pendapatan) baru yang mungkin lebih menjanjikan yang tidak bisa ditemui di sistem transaksi tradisional.
- b) Dapat meningkatkan market exposure (pangsa pasar).
- c) Menurunkan biaya operasional (operating cost).
- d) Melebarkan jangkauan (global reach).
- e) Meningkatkan customer loyalty.
- f) Meningkatkan supplier management.
- g) Memperpendek waktu produksi.
- h) Meningkatkan value chain (mata rantai pendapatan).

Dampak negatifnya, yaitu :

- a) Kehilangan segi finansial secara langsung karena kecurangan. Seorang penipu mentransfer uang dari rekening satu ke rekening lainnya atau dia telah mengganti semua data finansial yang ada.

- b) Pencurian informasi rahasia yang berharga. Gangguan yang timbul bisa menyingkap semua informasi rahasia tersebut kepada pihak-pihak yang tidak berhak dan dapat mengakibatkan kerugian yang besar bagi si korban.
- c) Kehilangan kesempatan bisnis karena gangguan pelayanan. Kesalahan ini bersifat kesalahan non-teknis seperti aliran listrik tiba-tiba padam.
- d) Penggunaan akses ke sumber oleh pihak yang tidak berhak. Misalkan seorang hacker yang berhasil membobol sebuah sistem perbankan. Setelah itu dia memindahkan sejumlah rekening orang lain ke rekeningnya sendiri.
- e) Kehilangan kepercayaan dari para konsumen. Ini karena berbagai macam faktor seperti usaha yang dilakukan dengan sengaja oleh pihak lain yang berusaha menjatuhkan reputasi perusahaan tersebut.
- f) Kerugian yang tidak terduga. Disebabkan oleh gangguan yang dilakukan dengan sengaja, ketidakjujuran, praktek bisnis yang tidak benar, kesalahan faktor manusia, kesalahan faktor manusia atau kesalahan sistem elektronik.

### **2.6.8 kategori utama dari *E-Commerce***

Di bawah ini adalah kategori-kategori *e-commerce* yaitu :

a. *Business-to-consumer (B2C)*

Di dalam transaksi *B2C*, transaksi *online* dibuat antara bisnis dan konsumen. Transaksi ini meliputi transaksi penjualan dengan pembeli-pembeli individu.

b. *Business-to-business (B2B)*

Di dalam *B2B*, transaksi online dilakukan antara bisnis dan bisnis yang lain. Partisipannya meliputi bisnis atau organisasi lainnya.

c. *Consumer-to-consumer (C2C)*

Pada kategori ini, transaksi terjadi antara konsumen yang satu dengan konsumen yang lainnya.

d. *Collaborative commerce*

*Collaborative commerce* merupakan suatu aplikasi dari IOS (*Interorganizational Information System*) untuk kolaborasi elektronik antara rekan bisnis dan antara pegawai perusahaan atau organisasi.

e. *Nonbusiness E-Commerce*

Dewasa ini makin banyak lembaga non-bisnis seperti lembaga akademis, organisasi nirlaba, organisasi keagamaan, organisasi sosial, dan lembaga-lembaga pemerintahan yang menggunakan berbagai tipe *E-Commerce* untuk mengurangi biaya (misalnya, memperbaiki *purchasing*) atau untuk meningkatkan operasi dan layanan publik. Contohnya *Social Security Online*.

f. *Intrabusiness (Organizational) E-Commerce*

Yang termasuk dalam kategori ini adalah semua aktivitas intern organisasi, biasanya dijalankan di internet, yang melibatkan pertukaran barang, jasa atau informasi. Aktivitas yang tercakup dapat beragam tingkatannya, mulai penjualan produk perusahaan ke pekerja, hingga pelatihan secara online dan pemangkasan biaya.

## **2.7 Internet Marketing**

### **2.7.1 Pengertian Internet Marketing**

Internet marketing terdiri dari dua kata, internet dan marketing. Internet menurut Wikipedia “Internet adalah jaringan komputer skala dunia yang memungkinkan orang-orang untuk saling berhubungan menggunakan berbagai layanan, seperti *email*, *chat online*, *transfer file*, dan halaman *web (WWW)*”. (Ali Zaki, 2008)

Sementara kata kedua adalah marketing. Marketing atau dalam bahasa indonesia disebut pemasaran ini. Marketing adalah sebuah proses bertujuan mengetahui keinginan konsumen, mengalokasikan produk atau jasa yang sesuai dengan keinginan tersebut, membuat konsumen tertarik dengan produk atau jasa tersebut.

Dari kedua arti diatas terciptalah internet marketing yaitu proses marketing/ pemasaran barang atau jasa melalui internet, kata kunci internet marketing adalah pada kata melalui internet. Hal inilah yang membedakan internet marketing dengan marketing konvensional, yaitu pada penggunaan media internet guna mendukung proses marketing. internet marketing merupakan sesuatu yang luar biasa karena adanya kata internet didalamnya. Internet memungkinkan proses

marketing lebih efektif, responya lebih cepat, namun dengan biaya yang lebih murah.

### **2.7.2 Model Bisnis dari Internet Marketing**

Ada banyak model bisnis yang lazim diterapkan di internet marketing, penggunaan model-model bisnis yang berbeda-beda ini tergantung kepada tujuan yang diinginkan di internet marketing tersebut. Di antaranya adalah : (Ali Zaki, 2008).

- a. *E-commerce*, adalah sebuah barang langsung dijual menggunakan internet, baik untuk konsumen (*B2C*) maupun untuk bisnis (*B2B*).
- b. Publishing, internet digunakan untuk menampilkan barang saja.
- c. Periklanan, website digunakan untuk menampilkan iklan-iklan saja, sementara transaksi masih dilakukan secara manual.

### **2.7.3 Keuntungan Internet Marketing**

Internet marketing menawarkan beberapa keuntungan dibandingkan dengan marketing konvensional. Baik ditinjau dari efektifitas maupun efisiensi. Berikut merupakan beberapa keuntungannya adalah : (Ali Zaki, 2008).

- a. Keuntungan utama internet marketing adalah skalanya yang sangat luas. Ukuran untuk mengukur jangkauan internet marketing tidak hanya berupa skala kecamatan, kota, propinsi, bahkan negara. Jangkauan internet marketing adalah global.
- b. Keuntungan lainnya adalah banyaknya informasi yang bisa disampaikan melalui internet.
- c. Karena informasi yang disampaikan bisa lebih banyak, maka konsumen bisa memperoleh gambaran yang lebih baik tentang perusahaan ataupun produk.
- d. Lebih hemat biaya. Dengan menggunakan internet marketing, perusahaan bisa mengurangi biaya untuk menyewa tenaga pemasaran.

## 2.8 Konsep Dasar Web

Konsep dasar web adalah sebuah konsep yang digunakan untuk membangun sebuah web.

### 2.8.1 Aplikasi Web

Pada awalnya aplikasi *Web* hanya dibangun hanya menggunakan bahasa yang disebut dengan *HTML (hyperText Markup Language)* dan *protokol* yang digunakan *HTTP (HyperText Transfer Protokol)*. Pada perkembangan berikutnya sejumlah *Script* dan objek dikembangkan untuk memperluas kemampuan *HTML*. *Script-script* tersebut antara lain *ASP (Active Server Page)* dan *PHP* contoh yang berupa objek antara lain *Applet (Java)*.

Aplikasi *Web* dapat dibagi menjadi dua yaitu *web* yang bersifat statis dan *web* yang bersifat dinamis, *web* statis dibentuk dengan menggunakan *HTML* saja, kekurangan aplikasi ini terletak pada keharusan untuk memelihara program secara terus-menerus untuk mengikut setiap perubahan yang terjadi. Kelemahan ini diatasi dengan model *Web* dinamis.

Dengan adanya *Web* dinamis perubahan informasi dalam halaman-halaman *Web* dapat ditangani melalui perubahan data, bukan melalui perubahan program. Sebagai implementasinya aplikasi *Web* dapat dikoneksi ke *database* arsitektur aplikasi *Web*, *client* berinteraksi dengan *Web master*, secara internal *Web server* berkomunikasi dengan *Middleware* (perangkat lunak penerjemah kode-kode program) dan *middleware* inilah yang berhubungan dengan *database*. (Andi, 2002).

### 2.8.2 Teknologi Web

Dari sisi teknologi yang digunakan untuk membentuk *Web* dinamis, terdapat dua macam pengelompokan yaitu teknologi pada sisi *client (Client-side Teknologi)* dan teknologi pada sisi *server (Server-side Teknologi)*

### 2.8.2.1 Teknologi Web Pada Sisi Client

Teknologi pada sisi *client* diimplementasikan dengan mengirimkan kode perluasan HTML atau program tersendiri dan HTML ke *client*. *Client* tersebut yang bertanggung jawab dalam melakukan proses terhadap seluruh kode yang diterima.

Kelemahan pendekatan seperti ini adalah terdapat kemungkinan bahwa *browser* pada *client* tidak mendukung fitur kode perluasan HTML. Misalnya kode *VBScript* yang dilekatkan pada kode HTML tidak berfungsi jika *browser* yang digunakan pada sisi *client* tidak mendukung. Kelebihannya memungkinkan penampilan yang bersifat dimanis ataupun animasi gambar yang melibatkan *event mouse*.

### 2.8.2.2 Teknologi Web Pada Sisi Server

Teknologi *Web* pada sisi *server* memungkinkan pemrosesan kode di dalam *server* sehingga kode yang sampai pada pemakai berbeda dengan kode asli pada *server*.

Beberapa teknologi yang berjalan di *server* adalah:

1. *PHP*, *PHP* merupakan *script* yang berjalan di *server* dan sangat populer di lingkungan Linux, saat ini *PHP* dapat berjalan di lingkungan berbagai *platform* mulai dari UNIX hingga *Windows*.
2. *ASP (Active Server Page)*, *ASP* merupakan teknologi di sisi *server* yang paling banyak digunakan di lingkungan *Windows*, saat ini *ASP* juga diusahakan untuk berjalan pada *platform* selains *Windows*. Teknologi lanjutan dari *ASP* adalah *ASP.Net*
3. *Server-Side JavaScript, Sever-Side JavaScript (SSJS)* merupakan buatan Netscape yang ditujukan untuk menyaingi *ASP (Active Server Page)*. Dalam prakteknya *script* ini kurang populer.
4. *Java Servlets dan JavaServer Page (JSP)*, Baik *JavaServlets* maupun *JavaServer Page* menggunakan bahasa Java, oleh karena itu pemanfaatan fitur Java ini dapat digunakan untuk membuat program yang canggih.



Meskipun bahasa Java tidak begitu sulit dibanding bahasa C namun pemakaian program *Java Servlets* dan JSP masih kalah dengan ASP dan PHP

5. *Common Gateway Interface (CGI)*, Pada awalnya, teknologi yang umum digunakan untuk menyajikan data yang bersifat dinamis di lingkungan *Web* adalah CGI. Pada prinsipnya CGI dapat ditulis dengan menggunakan bahasa apa saja tetapi yang populer adalah Perl. Kelemahan CGI terletak pada penciptaan proses sebanyak permintaan dari *client*. Jika terjadi permintaan yang cukup banyak maka cenderung akan melumpuhkan *server*.
6. *Proprietary Web server API*, *micorsoft* dan *Netscape* menciptakan API (*Aplication Programming Interface*) yang ditujukan untuk membuat aplikasi *Web* dinamis. Pustaka dari *Microsoft* dinamakan *ISAPI (Internet Server API)* dan pustaka dari *Netscafe* disebut *NSAPI (Netscafe Server API)*. Kedua API ini dimaksudkan untuk mengatasi kelemahan CGI, karena keduanya dirancang untuk menciptakan sebuah proses saja, sekalipun terdapat banyak permintaan.