

BAB 2

LANDASAN TEORI

2.1 Teori-teori umum

2.1.1 Internet

2.1.1.1 Pengertian Internet

Pengertian menurut *Strauss, El-Ansary, Frost* (2003, p8) Internet adalah seluruh jaringan yang saling terhubung satu sama lain. Beberapa komputer-komputer dalam jaringan ini menyimpan *file*, seperti halaman web, yang dapat diakses oleh seluruh jaringan komputer.

Menurut *O'Brien* (2003, p10) Internet merupakan jaringan komputer yang berkembang pesat dari jutaan bisnis, pendidikan, dan jaringan pemerintahan yang saling berhubungan dengan jumlah penggunanya lebih dari 200 negara.

Menurut *Allan* (2005, p12) Internet adalah sekumpulan jaringan komputer yang saling terhubung secara fisik dan memiliki kemampuan untuk membaca dan menguraikan protokol komunikasi tertentu yang disebut *Internet Protocol (IP)* dan *Transmission Control Protocol (TCP)*. Protokol adalah spesifikasi sederhana mengenai bagaimana komputer saling bertukar pikiran.

Berdasarkan tiga pengertian internet diatas dapat disimpulkan internet merupakan seluruh jaringan yang saling terhubung secara fisik dan memiliki kemampuan untuk membaca dan menguraikan protokol komunikasi dengan jumlah pengguna lebih dari 200 negara.

2.1.1.2 Sejarah Internet

Pada awalnya internet merupakan jaringan komputer yang dibentuk oleh Departemen Pertahanan Amerika Serikat di tahun 1969, melalui proyek ARPA yang disebut ARPANET (*Advanced Research Proyek Agency Net*), dimana mereka mendemostrasikan bagaimana dengan hardware dan software komputer yang berbasis UNIX, kita bisa melakukan komunikasi dalam jarak yang tidak terhingga melalui saluran telepon. Proyek ARPANET merancang bentuk jaringan, kehandalan, seberapa besar informasi dapat dipindahkan, dan akhirnya semua standar yang mereka tentukan menjadi cikal bakal untuk pembangunan protokol baru yang sekarang dikenal sebagai TCP/IP (*Transfer Control Protocol / Internet Protocol*).

Tujuan awal dibangunnya proyek itu adalah untuk keperluan militer. Pada saat itu Departemen Pertahanan Amerika Serikat (US Department Of Defense's) membuat sistem jaringan komputer yang tersebar dengan menghubungkan komputer di daerah-daerah vital untuk mengatasi masalah bila terjadi serangan nuklir dan untuk menghindari terjadinya informasi terpusat, yang apabila terjadi perang dapat mudah dihancurkan.

Pada mulanya ARPANET hanya menghubungkan 3 situs saja yaitu, *Stanford Research Institute, University of California di Santa Barbara, University of Utah*, dimana mereka membentuk satu jaringan terpadu di tahun 1969, dan secara umum ARPANET diperkenalkan pada bulan Oktober 1972. Tidak lama kemudian proyek ini berkembang pesat di seluruh daerah, dan semua

universitas di negara tersebut ingin bergabung, sehingga membuat ARPANET kesulitan untuk mengaturnya.

Oleh sebab itu ARPANET dipecah menjadi dua, yaitu “MILNET” untuk keperluan militer dan “ARPANET” baru yang lebih kecil untuk keperluan non-militer seperti, universitas-universitas. Gabungan kedua jaringan akhirnya dikenal dengan nama DARPA internet, yang kemudian disederhaakan menjadi internet.

2.1.1.3 Keistimewaan dan keunggulan Internet

Menurut Ustadiyanto (2003, p15) internet merupakan media yang memiliki keistimewaan dan keunggulan. Berikut ini adalah beberapa contoh keistimewaan dan keunggulan internet dibandingkan media-media lainnya.

1. Efisien

Bila ditinjau secara mendalam, internet sesungguhnya lebih efisien dan lebih ekonomis dari media lainnya.

2. Tanpa Batasan

Internet tidak mengenal batas negara atau benua, jadi dalam waktu yang relatif singkat informasi dapat disebarkan ke seluruh pelosok dunia.

3. 24 jam non stop

Intenet adalah media yang berlangsung sepanjang waktu, ini berarti kapanpun pengguna internet memerlukan informasi dapat langsung mengunjungi situs yang diinginkan untuk mencari informasi tersebut

4. Interaktif

Hampir seluruh media interaktif yang telah disepakati secara meluas merupakan

media yang menyediakan informasi satu arah, dimana seseorang tidak memiliki peluang untuk ikut andil didalamnya.

Meskipun terdapat media yang memperbolehkan ikut andil, namun memerlukan waktu yang relatif lama. Sebaliknya, internet merupakan media interaktif yang memungkinkan seorang user untuk ikut berpartisipasi kapanpun

2.1.1.4 Layanan-Layanan Internet

Seiring dengan perkembangan yang terus meningkat, kini di *internet* telah tersedia berbagai macam layanan, antara:

1. *Audio / video Streaming*

Suatu teknologi yang memungkinkan suatu *file* untuk dapat langsung digunakan sebelum di-download seluruhnya, contoh *RealPlayer*.

2. *Email (Electronic Mail)*

E-mail digunakan untuk mengirim pesan juga dapat menyertakan file yang dialamatkan seorang *user* pada sebuah *mailserver*.

3. *FTP (File Transfer Protocol)*

FTP memungkinkan sebuah *local* computer dengan menggunakan *file* FTP *client* untuk menghubungi FTP Server yang ada pada sebuah *remote computer* agar dapat saling bertukar *file*.

4. *Gopher*

Layanan yang menyediakan informasi berbasis teks, untuk mencari informasi pada *gopher* dapat digunakan VERONICA (*Very Easy Rodent Oriented Net-Wide Index to Computerized Archive*)

5. *Instant Messenger*

Program yang memungkinkan penggunanya untuk berkirim pesan secara *on-line person-to-person*, contoh : ICQ

2.1.2 World Wide Web (WWW)

2.1.2.1 Pengertian WWW

Menurut Meleod (2001, P75), pada tahun 1989, Tim Berners dan Lee menemukan cara yang lebih baik bagi fisikawan untuk berkomunikasi. Idenya adalah menggunakan *Hypertext* dokumen-dokumen dalam bentuk elektronik yang saling dikaitkan dengan cara tertentu. Ide itu menjadi kenyataan pada pertengahan tahun 1992 dalam bentuk World Wide Web, dan sejak itu jauh melampaui harapan awalnya. Bukan hanya sekedar menangani teks, tapi juga mampu menyimpan dan mengambil *Hypermedia*, multimedia yang terdiri dari teks, grafik, *audio*, dan *video*. *World Wide Web*, juga disebut *web*, *WWW* yang berarti ruang informasi dan dapat diambil melalui suatu skema alamat yang unik.

2.1.2.2 Istilah-istilah dalam WWW

Menurut McLeod (2004,p65) banyak istilah yang secara normal dikaitkan dengan *internet* sebenarnya berhubungan dengan *web*,diantaranya:

- a). *Website* ini mengacu pada suatu komputer yang dikaitkan ke internet yang berisi *hypermedia* yang dapat diakses oleh komputer lain di jaringan melalui suatu *hypertext link*.
- b). *hypertext link* ini mengacu pada suatu petunjuk yang terdiri dari text atau grafik yang digunakan untuk mengakses *hypertext* yang disimpan di Website. Teks ini biasanya digarisbawahi dan ditampilkan dalam warna biru. Jika kursor ditempatkan diatasnya, bentuk kursor ini berubah

menjadi tangan dengan jari yang menunjuk.

c). *Web page* ini mengacu pada suatu file *hypermedia* yang disimpan di suatu website, yang diidentifikasi oleh suatu alamat yang unik.

d). *Home page* ini mengacu pada halaman pertama dari website. Halaman-halaman lain di site tersebut dapat dicapai dari *home page*.

e). URL(*Universal Resource Locator*). Ini mengacu pada alamat suatu *web page* diucapkan "earl".

f). *Protocol*. Satu set standar yang mengatur komunikasi data. HTTP adalah *protocol* untuk *hypertext*. Dan huruf-huruf itu adalah singkatan dari *hypertext Transport Protocol*. Nama *Protocol* dalam huruf kecil, dan diikuti oleh titik dua(:) dan dua garis miring(/).

2.1.3 Pengertian pemasaran

Menurut Kotler (2002, p9), pemasaran adalah suatu proses sosial yang didalamnya individu dan kelompok mendapatkan apa yang mereka butuhkan dan inginkan dengan menciptakan, menawarkan, dan secara bebas mempertukarkan produk yang bernilai dengan pihak lain .

Menurut McLeod (2001, p449), pemasaran terdiri dari kegiatan perorangan dan organisasi yang memudahkan dan mempercepat hubungan pertukaran yang memuaskan dalam hubungan yang dinamis melalui penciptaan, pendistribusian, promosi, dan penentuan harga barang, jasa dan gagasan.

Menurut Ranguti (2002, p48), pemasaran adalah suatu proses kegiatan yang dipengaruhi oleh berbagai faktor sosial, budaya, politik, ekonomi dan manajerial . Akibat dari faktor-faktor tersebut adalah masing-masing individu

maupun kelompok mendapatkan kebutuhan-kebutuhan dan keinginan dengan menciptakan, menawarkan, dan menukarkan produk yang memiliki komoditas .

2.1.3.1 Tujuan Pemasaran

Tujuan pemasaran menurut Peter Drucker adalah membuat kegiatan menjual berjalan lancar, yaitu untuk mengetahui dan memahami para pelanggan dengan baik sehingga produk atau jasa yang dihasilkan cocok dengan mereka yang dapat terjual dengan sendirinya . Idealnya, pemasaran harus menghasilkan pelanggan yang siap membeli .

2.1.3.2 Pengertian manajemen pemasaran

Menurut Koetler dan Armstrong (2001, p18), manajemen pemasaran adalah didefinisikan sebagai analisis, perencanaan, implementasi, dan pengendalian dari program-program yang dirancang untuk menciptakan, membangun dan memelihara pertukaran dengan pembeli, sasaran yang menguntungkan perusahaan untuk mencapai tujuan perusahaan .

Manajemen pemasaran adalah proses perencanaan dan pelaksanaan pemikiran penetapan harga, promosi serta penyaluran gagasan barang dan jasa untuk menciptakan pertukaran yang memenuhi sasaran-sasaran individu dan organisasi. (Kotler, 2002, p9)

Berdasarkan Teori diatas dapat disimpulkan bahwa manajemen pemasaran adalah suatu proses analisis perencanaan dan implementasi yang mempunyai tujuan menciptakan pertukaran yang memenuhi sasaran-sasaran individu dan menciptakan tujuan perusahaan

2.1.3.3 Konsep Pemasaran

Menurut Kotler dan Armstrong (2001, p 8-17), konsep pemasaran inti terdiri dari :

1. Kebutuhan, keinginan dan permintaan

Kebutuhan manusia (*needs*) adalah pernyataan dari perasaan kekurangan. Keinginan (*wants*) adalah kebutuhan manusia yang dibentuk oleh budaya dan kepribadian seseorang. Ketika didukung oleh daya beli, keinginan berubah menjadi permintaan (*demands*).

2. Produk dan jasa

Produk (*products*) adalah segala sesuatu yang dapat ditawarkan ke pasar untuk memuaskan kebutuhan atau keinginan pelanggan . Selain barang nyata, produk meliputi jasa, yang merupakan aktivitas atau manfaat yang ditawarkan untuk dijual, yang pada dasarnya tidak nyata dan tidak berakibat pada kepemilikan apapun.

3. Nilai, kepuasan dan menu

Nilai pelanggan (*customer value*) merupakan selisih antara nilai yang diperoleh pelanggan dengan memiliki dan menggunakan suatu produk, dengan biaya yang dikeluarkan (*cost*) untuk memperoleh produk tersebut . Kepuasan pelanggan (*customer satisfaction*) bergantung pada perkiraan kinerja produk dalam memberikan nilai, relatif pada terhadap harapan pembeli. Kepuasan pelanggan berkaitan dengan kualitas. Manajemen mutu total (*total quality management*) adalah suatu program yang dirancang untuk memperbaiki mutu produk, jasa, dan proses pemasaran secara terus-menerus .

4. Pertukaran, transaksi, dan hubungan pemasaran

Pertukaran (*exchange*) adalah suatu tindakan untuk memperoleh obyek yang diharapkan dari seseorang dengan menawarkan sesuatu sebagai pengganti . Transaksi (*transaction*) adalah suatu perdagangan yang melibatkan paling sedikit dua bentuk nilai, persetujuan mengenai kondisi, persetujuan mengenai waktu dan persetujuan mengenai kondisi, persetujuan waktu dan persetujuan mengenai mengenai tempat . Transaksi pemasaran merupakan bagian dari gagasan yang lebih besar mengenai tempat . Transaksi pemasaran merupakan bagian dari gagasan yang lebih besar mengenai hubungan pemasaran (*relationship management*). Hubungan pemasaran yaitu proses menciptakan , memelihara, dan meningkatkan hubungan timbal balik dengan pelanggan dan pihak lain yang berkepentingan .

5. Pasar

Tempat dimana penawaran dan permintaan bertemu untuk melakukan suatu pertukaran atau transaksi

6. Pemasaran

Pemasaran(*marketing*) berarti mengelolah pasar untuk menghasilkan pertukaran dan hubungan dengan tujuan menciptakan nilai dan memuaskan kebutuhan akan kegiatan.

2.1.3.4 Riset Pemasaran

Riset pemasaran(*marketing research*) adalah kegiatan penelitian di bidang pemasaran yang dilakukan secara sistematis mulai dari perumusan masalah,tujuan masalah, pengumpulan data, pengolahan data,dan interpersi

hasil penelitian(Rangkuti2002,p1).Tujuan riset pemasaran adalah:

- Mendapatkan informasi yang akurat sehingga dapat menjelaskan kenyataan yang ada secara objektif.
- Bebas dari pengaruh keinginan pribadi(political biases).”Find it and tell it like it is.

2.1.3.5 Proses Pemasaran

Menurut kotler dan Armstrong(2001,p67),proses pemasaran adalah proses yang (1)Menganalisis peluang pemasaran; (2)Menyeleksi pasar sasaran; (3)Mengembangkan bauran pemasaran; dan(4)Mengatur usaha pemasaran.

Rencana strategis mendefinisikan misi dan tujuan perusahaan secara keseluruhan.Dalam setiap unit bisnis, Pemasaran memainkan peran dalam membantu pencapaian seluruh tujuan strategis.

Konsumen sasaran berada dibagian tengah. Perusahaan mengidentifikasi seluruh pasaran, membaginya kedalam segment yang lebih kecil, memilih segment yang paling menjanjikan, dan memusatkan perhatian dalam pelayanan serta pemuasan segment ini. Perusahaan merancang bauran pemasaran yang disusun oleh faktor-faktor dibawah kendali produk,harga,lokasi,dan promosi.untuk mendapatkan bauran pemasaran yang terbaik dan dapat melaksanakan, perusahaan turut serta dalam analisis pemasaran, perancangan,implementasi, dan pengawasan. Melalui kegiatan-kegiatan ini, perusahaan melakukan pengamatan dan penyesuaian terhadap lingkungan pemasaran.

2.2 Teori khusus

2.2.1 *e-Marketing*

2.2.1.1 Pengertian *e-Marketing*

Menurut Chaffey(2000,p6), *e-Marketing* adalah aplikasi dari *internet* dan teknologi digital lain yang berhubungan untuk mencapai tujuan-tujuan marketing.

Menurut mohammed(2003,p4), *e-Marketing* adalah suatu proses membangun dan memelihara hubungan dengan pelanggan melalui aktivitas online dalam memfasilitasi pertukaran atas ide, produk, dan layanan agar memuaskan kedua belah pihak.

E-Marketing adalah segala usaha yang dilakukan untuk melakukan pemasaran suatu produk atau jasa melalui atau menggunakan media internet atau jaringan www.. Kata "e" dalam *e-Marketing* ini berarti elektronik(eletronic) yang artinya kegiatan pemasaran yang di maksud dilaksanakan secara elektronik lewat internet atau jaringan cyber.

Kegiatan pemasaran melalui *internet* umumnya meliputi atau berkisar pada hal-hal yang berhubungan pada pembuatan produk periklanan,pencarian prospek atau pembeli dan penulisan kalimat-kalimat pemsaran atau copywriting.Pemasaran melalui *internet* atau *e-Marketing* ini secara umum meliputi kegiatan pembuatan desain web(web design), periklanan dengan menggunakan banner, promosi perusahaan lewat mesin pencari informasi (search engine), surat elektronik atau *e-mail*, periklanan lewat e-mail (*e-mail advertising*), pemasaran afiliasi (*affiliate marketing*), advertensi interaktif

(*interactive advertsing*), dan lain-lain.

2.2.1.2 Strategi *e-marketing*

Strategi *e-marketing* yang paling umum digunakan adalah model "CTPM" yang dipopulerkan oleh Ken Evoy, yang meliputi :

1.Content

Menyediakan segala informasi yang dibutuhkan seputar jasa atau produk yang akan dipasarkan .

2.Traffic

Bagaimana caranya untuk mendatangkan pengunjung untuk melihat jasa atau produk yang dihasilkan .

3.Pre-sell

Membentuk kesan jasa atau produk yang akan dipasarkan

4.Monetize

Mendapatkan penjualan dari jasa atau produk yang dipasarkan

2.2.2 E-Commerce

2.2.2.1 Pengertian *E-Commerce*

E-Commerce atau yang sering juga disebut dengan *Internet Commerce* atau *Ecom* pada dasarnya mempunyai makna yang sama, yang berarti suatu cara bagi seorang konsumen untuk dapat membeli barang yang diinginkan secara *online* melalui jaringan internet. Definisi dari *E-Commerce* menurut Kalakota dan Whinston (1997) dapat ditinjau dalam 4 perspektif berikut:

- Dari perspektif komunikasi, *E-Commerce* adalah pengiriman barang, layanan, informasi, atau pembayaran melalui jaringan komputer atau melalui peralatan

elektronik lainnya.

- Dari perspektif proses bisnis, *E-Commerce* adalah aplikasi dari teknologi yang menuju otomatisasi dari transaksi bisnis dan aliran kerja.
- Dari perspektif layanan, *E-Commerce* merupakan suatu alat yang memenuhi keinginan perusahaan, konsumen, dan manajemen untuk memangkas biaya layanan (*service cost*) ketika meningkatkan kualitas barang dan meningkatkan kecepatan layanan pengiriman.
- Dari perspektif *online*, *E-Commerce* menyediakan kemampuan untuk membeli dan menjual barang ataupun informasi melalui internet dan sarana *online* lainnya.

2.2.2.2 Penggolongan *E-Commerce*

Penggolongan *E-Commerce* berdasar sifat transaksinya, antara lain:

- *Business-to-Business (B2B)*

Seluruh peserta dalam tipe ini adalah orang-orang bisnis atau organisasi/perusahaan lain. Kebanyakan *E-Commerce* yang diterapkan saat ini adalah tipe *B2B*(Cunningham, 2001). Transaksi *B2B* meliputi transaksi *IOS* dan transaksi *e-market* antara dua atau lebih organisasi/perusahaan.

- *Business-to-Consumer (B2C)*

Transaksi ini meliputi transaksi eceran(*retail*) dengan pembeli perorangan. Model bisnis ini sering pula disebut *e-tailing*. Contohnya, *Amazon*, *Dell*, *Compaq*.

- *Consumer-to-Consumer (C2C)*

Dalam tipe ini, seorang konsumen menjual secara langsung ke konsumen lainnya. Mengiklankan jasa pribadi di internet serta menjual pengetahuan dan

keahlian merupakan contoh dari C2C. Banyak per-orangan yang menggunakan intranet dan jaringan organisasi untuk mengiklankan barang-barang yang akan dijual atau bisa juga menawarkan beragam jasa. Contohnya adalah *eBay.com*, yang merupakan suatu perusahaan lelang.

- *People-to-People(P2P)*

Tipe transaksi ini adalah tipe khusus dari C2C dimana orang menukarkan CD, video, perangkat lunak(*software*), dan barang lain. Contohnya adalah *Napster.com*.

- *Consumer-to-Business (C2B)*

Kategori ini meliputi per-orangan/individu yang menggunakan internet untuk menjual barang atau jasa ke perusahaan, per-orangan yang mencari penjual, berinteraksi dengan mereka, dan melakukan transaksi secara *online*. Contohnya adalah *Priceline.com*.

- *Intrabusiness (Organizational) E-Commerce*

Kategori ini meliputi semua aktivitas di dalam perusahaan, biasanya dijalankan dengan menggunakan intranet atau portal yang terhubung, yang melibatkan pertukaran barang, jasa atau informasi antara beberapa unit dan individu dalam organisasi tersebut. Aktivitasnya tercakup dalam beragam tingkatan, mulai penjualan produk perusahaan ke pekerja, hingga pelatihan secara *online*.

- *Business-to-employees(B2F)*

Tipe ini merupakan bagian dari kategori *intrabusiness* dimana perusahaan mengantarkan jasa, informasi, atau barang kepada pekerja perorangan.

- *Government-to-citizens(G2C)*

Dalam kategori ini pemerintah membeli atau menjual barang, jasa, dan informasi kepada pelaku bisnis atau kepada perorangan.

- *Exchange-to-Exchange(E2E)*

E2E merupakan sistem resmi yang menghubungkan pertukaran dalam *E-Commerce*.

- *Collaborative Commerce*

Merupakan kolaborasi dari *IOS* untuk penggabungan elektronik antara rekan bisnis dan antara pekerja perusahaan.

- *Mobile Commerce*

Ketika *E-Commerce* mengambil alih lingkungan *wireless*, maka *mobile commerce* ini pun berkembang pesat. Dimana para pelanggan telepon selular dapat melakukan transaksi bisnis menggunakan *E-Commerce* melalui telepon selularnya dalam kategori ini.

- *Nonbusiness E-Commerce*

Kategori ini meliputi kegiatan non bisnis, misalnya lembaga akademis, organisasi keagamaan, organisasi sosial yang menggunakan teknologi *E-Commerce* untuk dapat melakukan penghematan biaya atau untuk meningkatkan operasi dan layanan umum.

2.2.2.3 Standar Teknologi *E-Commerce*

Di samping berbagai standar yang ada di internet, *E-Commerce* juga menggunakan standar sendiri yang umumnya digunakan dalam transaksi *B2B* (*Business-to-Business*), antara lain sebagai berikut:

- *Electronic Data Interchange (EDI)*

EDI dibuat pada awal tahun 70-an dan saat ini telah digunakan oleh lebih dari 1000 perusahaan *Fortune* di Amerika Serikat. *EDI* adalah sebuah standar struktur dokumen yang dirancang untuk memungkinkan organisasi besar mengirimkan informasi melalui jaringan khusus (*private network*).

- *Open Buying on the Internet (OBI)*

OBI adalah sebuah standar yang dibuat oleh *Internet Purchasing Roundtable* yang akan menjamin berbagai system *E-Commerce* dapat berbicara/berhubungan satu sama lain. *OBI* dikembangkan oleh konsorsium *OBI* yang didukung oleh perusahaan-perusahaan bidang teknologi, seperti *Microsoft*, *Actra*, *InteliSys*, *Open Market* dan *Oracle*.

- *Open Trading Protocol (OTP)*

OTP dibuat untuk menstandarisasi berbagai aktivitas yang berkaitan dengan proses pembayaran, seperti perjanjian pembelian, resi pembelian dan pembayaran. *OTP* merupakan standar kompetitor *OBI*, yang dibangun oleh beberapa perusahaan seperti *IBM*, *Hitachi*, *SUN*, *AT&T*, *CyberCash*, dan *British Telecom*.

- *Open Profiling Standard (OPS)*

OPS merupakan sebuah standar yang didukung oleh *Microsoft* dan *Firefly*. *OPS* memungkinkan penggunanya untuk membuat sebuah profil pribadi yang sesuai dengan kesukaan masing-masing dan dapat di-*share* dengan *merchant*. Ide dibalik *OPS* adalah untuk menolong memroteksi privasi pengguna tanpa menutup kemungkinan melakukan transaksi informasi dalam proses pemasaran dan sebagainya.

- *Secure Socket Layer (SSL)*

SSL merupakan protokol yang didesain untuk membangun sebuah saluran yang aman ke *server*. *SSL* menggunakan teknik *enkripsi public key* untuk memroteksi data yang dikirimkan melalui internet. *SSL* dibuat oleh *Netscape*, tapi sekarang dipublikasikan di *public domain*.

2.2.2.4 Proses dan Arsitektur *E-Commerce*

Proses yang terjadi pada *E-Commerce* sebagai media pembelian dan penjualan secara elektronik adalah sebagai berikut:

1. *Show*, penjual menunjukkan produk dan layanannya di *website*. Dimana akan ditampilkan detail/spesifikasi dari produk yang dijual berikut harganya.
2. *Order*, setelah pembeli memilih produk yang diinginkan lewat katalog, pembeli memesan produk yang ada.
3. Verifikasi data pembeli, dapat juga disertakan nomor rekening ataupun kartu kredit.
4. Pembayaran yang dilakukan dari pihak pelanggan ke pihak penjual.
5. *Deliver*, pengiriman produk atau layanan yang dipesan.

Arsitektur dasar sebuah *E-Commerce* adalah arsitektur *client/server*. Proses *E-Commerce* akan berlangsung ketika terjadi interaksi antara komputer *client* dan *server*. Seorang pembeli akan berperan sebagai komputer *client* sedangkan pihak penjual akan berperan sebagai *server*. Penggunaan *E-Commerce* berbeda dengan penggunaan *Microsoft Word* yang hanya melibatkan sisi *client* saja, atau penggunaan mesin *VAX (Virtual Address Extension)* yang hanya melibatkan sisi *server*.

2.2.2.5 Manfaat *E-Commerce*

Dengan meluasnya perdagangan global, tidak cukup bagi sebuah perusahaan hanya dengan mengandalkan iklan dan selebaran untuk memajukan bisnis yang dijalankannya. Saat ini fenomena memperlihatkan paradigma yang semakin bergeser dari kekuatan ekonomi yang bertumpu pada pembuat (*manufacturer*) ke kekuatan pasar. Masyarakat *virtual* yang semakin banyak akan menggeser minat orang untuk melakukan bisnis melalui internet. Perusahaan yang pandai tidak akan melawan *trend* internet yang ada, tapi akan berusaha untuk melakukan inovasi guna mencapai dan memuaskan kebutuhan pelanggannya. Manfaat yang bisa diperoleh perusahaan di antaranya:

- *E-Commerce* memperluas daerah pemasaran hingga ke pasar nasional dan internasional. Dengan pengeluaran dana yang minim, perusahaan dapat dengan mudah dan cepat menemukan lebih banyak pelanggan, *supplier* yang lebih baik, dan rekan bisnis yang baru dan tepat dari seluruh dunia. Salah satu contohnya adalah *website Amazon* (<http://www.amazon.com>) yang menjual berbagai peralatan elektronik, buku secara *online*. Walaupun penjual berada jauh di benua Amerika namun pelanggannya yang berasal dari berbagai negara dapat membeli produknya hanya melalui komputer di rumah yang terkoneksi internet, dan hanya dalam beberapa hari barang yang dipesan telah sampai di tangan pelanggan.
- *E-Commerce* dapat mengurangi biaya pembuatan, pemrosesan, pendistribusian, penyimpanan, dan pencarian informasi yang menggunakan kertas.
- *E-Commerce* mengurangi waktu antara pengeluaran modal dan penerimaan barang dan jasa.

- *E-Commerce* memperkecil biaya telekomunikasi, karena internet lebih murah dibanding VANs (*Value Added Networks*).
- *E-Commerce* dapat mengurangi biaya administrasi hingga 80% atau lebih, mengurangi biaya *purchasing* hingga 5-10%, dan mengurangi *cycle time* hingga lebih dari 50%.
- *E-Commerce* memungkinkan interaksi yang lebih dekat dengan pelanggan, meskipun melalui media perantara. Ini adalah promosi yang lebih baik untuk *CRM* dan dapat meningkatkan loyalitas konsumen (*customer loyalty*).
- Layanan konsumen dan citra perusahaan menjadi lebih baik, dapat menemukan *partner* bisnis baru, menyederhanakan proses, mempersingkat waktu pemasaran, akses informasi menjadi cepat, mengurangi biaya transportasi, mengurangi biaya untuk kertas dan meningkatkan fleksibilitas.

E-Commerce tidak hanya mendatangkan manfaat yang menguntungkan bagi perusahaan namun juga memberikan manfaat bagi konsumen. Manfaat *E-Commerce* bagi konsumen antara lain:

- *E-Commerce* memungkinkan pelanggan untuk berbelanja atau melakukan transaksi lain sepanjang tahun selama 24 jam sehari dari manapun.
- *E-Commerce* menyediakan pilihan lebih banyak kepada pelanggan, sehingga pelanggan bisa memilih berbagai produk dari berbagai *vendor*.
- *E-Commerce* dapat membuat konsumen mendapatkan keperluannya dengan mengeluarkan biaya yang lebih rendah, karena konsumen yang berbelanja di toko cenderung mengeluarkan biaya lebih banyak karena tergoda untuk membeli barang-barang lain yang sebenarnya tidak diperlukan.

- Dalam beberapa produk, khususnya pada produk-produk digital, *E-Commerce* memungkinkan pengiriman yang sangat cepat.
- Pelanggan bisa mendapatkan informasi yang relevan dari detail produk dalam hitungan detik, bukan lagi hari atau minggu.
- *E-Commerce* memungkinkan untuk turut berpartisipasi dalam pelelangan maya (*virtual auction*). Hal ini memungkinkan penjual untuk menjual produknya dengan lebih cepat.
- *E-Commerce* memberikan tempat bagi pelanggan yang satu untuk berinteraksi dengan pelanggan lain di komunitas elektronik (*electronic community*) dan untuk bertukar pikiran serta berbagi pengalaman satu sama lain.

Manfaat *E-Commerce* bagi masyarakat antara lain:

- *E-Commerce* memungkinkan tiap individu untuk bekerja di dalam rumah dan mengurangi perjalanan ke luar rumah untuk berbelanja. Hal ini berakibat menurunkan arus kepadatan lalu lintas di jalan serta mengurangi polusi udara.
- *E-Commerce* memungkinkan sejumlah barang dijual dengan harga lebih murah, sehingga orang yang kurang mampu juga bisa membelinya dan akan meningkatkan taraf hidup mereka.
- *E-Commerce* memfasilitasi layanan umum, seperti perawatan kesehatan, pendidikan, dan layanan sosial pemerintah yang dapat diselenggarakan dengan biaya yang lebih rendah, dengan kualitas yang lebih baik. Sebagai contoh dokter yang bekerja di pedesaan dapat mengakses informasi dan teknologi sehingga mereka dapat melakukan perawatan untuk pasien dengan lebih baik.

2.2.2.6 Langkah Membangun Arsitektur *E-Commerce*

Menurut *Koontz* terdapat 6 langkah proses dalam membangun arsitektur *e-commerce*. Langkah-langkah tersebut antara lain:

1. Menetapkan tujuan dan visi bisnis

Tujuan dan visi bisnis sangat penting diketahui, sebagai langkah awal untuk menentukan proses dalam membangun arsitektur *e-commerce* selanjutnya.

2. Menetapkan informasi-informasi yang diperlukan

Informasi sangat penting untuk mencapai tujuan dalam membangun sebuah aplikasi *e-commerce*.

3. Menetapkan data yang dibutuhkan

Setelah mengetahui informasi apa yang harus diproses oleh aplikasi *e-commerce*, perlu ditentukan data dan informasi apa yang ingin diperoleh dari konsumen.

4. Menetapkan aplikasi yang digunakan

Dalam langkah ini, perlu ditentukan komponen atau modul aplikasi yang cocok dengan data yang dibutuhkan.

5. Menetapkan susunan secara teknis dari *e-commerce* yang akan dibuat

Dalam langkah ini, perlu menguji *hardware* yang spesifik dan *software* yang dibutuhkan untuk mendukung analisis pada langkah sebelumnya.

6. Menetapkan arsitektur organisasi

Arsitektur organisasi berhubungan dengan sumber daya manusia dan prosedur yang dibutuhkan dari langkah ke-1 sampai langkah ke-5.

2.2.3 Macromedia Dreamweaver MX

Macromedia Dreamweaver MX adalah sebuah aplikasi *web editor* , yang berbasis *WYSIWYG* (*What You See is What You Get* – Apa yang anda lihat adalah apa yang anda akan dapatkan). Dengan kemampuan ini maka orang-orang yang bahkan adalah pemula akan dengan mudah membangun sebuah halaman *web*, cukup dengan *klik mouse*. Bahkan membuat *web* memakai *Dreamweaver MX* hasilnya akan dapat langsung dilihat tanpa harus membuka *browser* terlebih dahulu, karena tampilannya akan selalu diperbaharui sesuai dengan perubahan kodenya.

Obyek-obyek yang disediakan dalam *Dreamweaver MX* antara lain:

- *Text*. Untuk menambahkan teks pada dokumen *web* yang dibuat, dapat langsung mengetikkannya pada dokumen atau mengcopy teks dari aplikasi lain dan *mempaste* teks tersebut ke dokumen *web*.
- *Image*. *Image* diperlukan untuk membuat halaman *web* tidak kaku dan menarik untuk dikunjungi, juga dapat mendukung informasi yang diberikan pada halaman *web*. Namun *image* membutuhkan waktu proses yang lebih lama untuk ditampilkan pada *browser* daripada teks biasa.
- *Table*. Digunakan untuk mengimplementasikan desain halaman *web* atau mempresentasikan data, dapat mengatur isi, terutama teks dan gambar pada sebuah halaman *web*.
- *Layer*. *Layer* digunakan sebagai kerangka *layout* desain halaman *web* atau untuk menganimasi tampilan halaman *web* sehingga menjadi lebih interaktif.
- *Frame*. Adalah sebuah daerah di jendela *browser* yang dapat menampilkan

dokumen *web* yang independen, berbeda dari apa yang ditampilkan di daerah lain dari jendela *browser*.

- *Link*. Berfungsi sebagai sarana navigasi antar halaman *web*. *Link* dapat menunjuk ke dokumen lain atau *file* lain, ke tempat tertentu pada satu dokumen (*link anchor*), *link e-mail* yang akan membuat sebuah *e-mail* kosong dengan alamat *e-mail* yang dituju sudah terisi, *link* kosong atau skrip.
- *Form*. Digunakan untuk memberi kesempatan pada pengunjung memberikan informasi, yang nantinya akan digunakan untuk proses tertentu. Misalnya *Text field*, *Check Box*, *Radio Button*, *Button*.
- *Media*. Merupakan sarana untuk menambahkan suara atau animasi pada halaman *web* secara cepat dan mudah. Misalnya animasi bertipe *Macromedia Flash* atau *Shockwave*, *Quick Time*.
- *Dynamic Content*. Umumnya berhubungan erat dengan proses membaca dan menyimpan data dari dan ke *database*.

2.2.4. Hypertext Markup Language (HTML)

2.2.4.1 Definisi dan Perkembangan HTML

HTML adalah *file teks* yang terdiri dari *tag-tag markup* yang biasa digunakan dalam semua *file-file* yang berjalan di atas *web browser*. Atau dengan kata lain, *HTML* merupakan serangkaian konstruksi standar yang dapat membangun halaman *web*. *HTML* merupakan pengembangan dari standar pemformatan dokumen teks, yaitu *Standard Generalized Markup Language (SGML)*.

HTML pertama kali dibuat oleh Tim Berners-Lee dan dipopulerkan pertama kali oleh *browser Mosaic*. Selama awal tahun 1990, *HTML* mengalami

perkembangan yang sangat pesat dan setiap pengembangannya *HTML* akan menambahkan kemampuan dan fasilitas yang lebih baik dari sebelumnya. Namun dalam perkembangannya, *HTML* tidak mengubah cara kerjanya. *HTML* 2.0 secara resmi dikeluarkan pada bulan November 1995 oleh *IETF (Internet Engineering Task Force)*. Kemudian *HTML* 3.0 (1995) memberikan kemampuan yang lebih dari versi sebelumnya. Dengan usaha dari *World Wide Web Consortium's HTML Working Group* pada tahun 1996, dihasilkan *HTML* 3.2. *HTML* versi terbaru adalah *HTML* 4.01 yang dikeluarkan secara resmi oleh *W3C* pada tanggal 24 April 1998. *File HTML* dapat dibuat dalam sebuah *teks editor*, misalnya yang sederhana di *Notepad*, dimana *editor notepad* ini dapat ditemukan pada hampir semua sistem operasi seperti *Windows 98 SE, ME, 2000, NT, XP*.

2.2.4.2. Elemen Dasar *HTML*

Elemen dasar yang terdapat dalam *HTML* antara lain:

a. Jenis dokumen (*HTML*)

HTML tag terdapat pada awal dan akhir dari *file HTML*. *Tag* ini adalah *tag* tentang *Web browser* dimana *HTML* pada dokumen berawal dan berakhir.

Strukturnya adalah sebagai berikut:

```
<html>
```

```
... ..
```

```
</html>
```

b. Judul *HTML*

Sebuah halaman *HTML* bisa memiliki judul dengan memanfaatkan pasangan *tag* `<TITLE>` dan `</TITLE>` yang diletakkan di dalam bagian *header* dari *HTML*.

Pada *teks* antara <TITLE> dan </TITLE>, akan ditampilkan pada *title bar* di *web browser*.

Contoh pemakaian judul:

```
<title>...judul web...</title>
```

c. *Heading*

Heading adalah sekelompok kata yang menjadi judul atau subjudul di dalam *file HTML*. *Head tag* berisi semua informasi *header* dokumen, keterangan umum, seperti judul dan sebagainya.

Strukturnya sebagai berikut:

```
<head>
```

```
.....
```

```
</head>
```

d. *Body*

Body diletakkan setelah struktur *Head*. Diantara *Body tag*, terdapat isian yang dapat ditampilkan dalam *browser window*. Di antara *Body tag* diletakkan semua teks, grafik, *link* dan sebagainya. Jadi di dalam *Body tag* ini mengandung isi dari halaman *HTML*.

Strukturnya adalah sebagai berikut:

```
<body>
```

```
.....
```

```
</body>
```

Jadi struktur *HTML* secara keseluruhan adalah sebagai berikut:

```
<html>
```

```

<head>

<title> Document Title </title>

</head>

<body>

</body>

</html>

```

2.2.5 Database

2.2.5.1 Pengertian Database

Menurut Connolly (2002, p14), *database* adalah kumpulan data yang secara logika saling berhubungan, memiliki deskripsi dari data, dirancang untuk memenuhi kebutuhan informasi. *Database* adalah sebuah tempat penyimpanan data yang besar yang dapat digunakan secara bergantian oleh berbagai departemen. Seluruh data yang ada terintegrasi dan memiliki sangat sedikit duplikasi. Secara umum definisi *database* dibagi menjadi dalam tiga fase, yaitu:

- Perancangan konseptual *database*.

Model data dibuat menggunakan informasi yang telah didokumentasikan oleh *user*. Perancangan konseptual *database* seluruhnya tidak tergantung pada detail implementasi, seperti tujuan *software* DBMS, aplikasi program, bahasa pemrograman dan pertimbangan lainnya.

- Perancangan logika *database*

Merupakan proses pembuatan model informasi yang digunakan pada perusahaan yang berdasarkan pada model data yang khusus, tetapi tidak bergantung pada bagian DBMS dan pertimbangan lainnya.

- Perancangan fisik *database*

Proses penggambaran penerapan *database* pada penyimpanan kedua, yang hubungan dasar, arsip organisasi dan indeks digunakan untuk menerima akses data secara efisien dan keseluruhan dan batasan serta ukuran keamanan.

2.2.5.2 Kunci (Key)

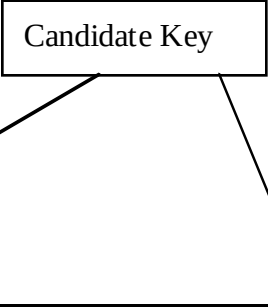
Kunci merupakan elemen record yang dipakai untuk menemukan record tersebut pada waktu akses atau bisa digunakan untuk identifikasi tiap record.

Jenis-jenis kunci :

- ✓ Superkey
- ✓ Candidat key
- ✓ Primary key
- ✓ Alternate key
- ✓ Foreign key
- ✓ External key

Superkey : kumpulan atribut dari suatu tabel yang dapat digunakan untuk mengidentifikasi entity atau record dari tabel tersebut secara unik

Candidate key : superkey dengan jumlah atribut minimal. Candidat key ini tidak boleh berisi atribut dari tabel yang lain



ID_Cus	Name	NoOfPay	Amount
112233	Tim	890	9000
112231	Kate	891	8000
112241	Tyson	895	10000

Table 2.1 tabel candidate key

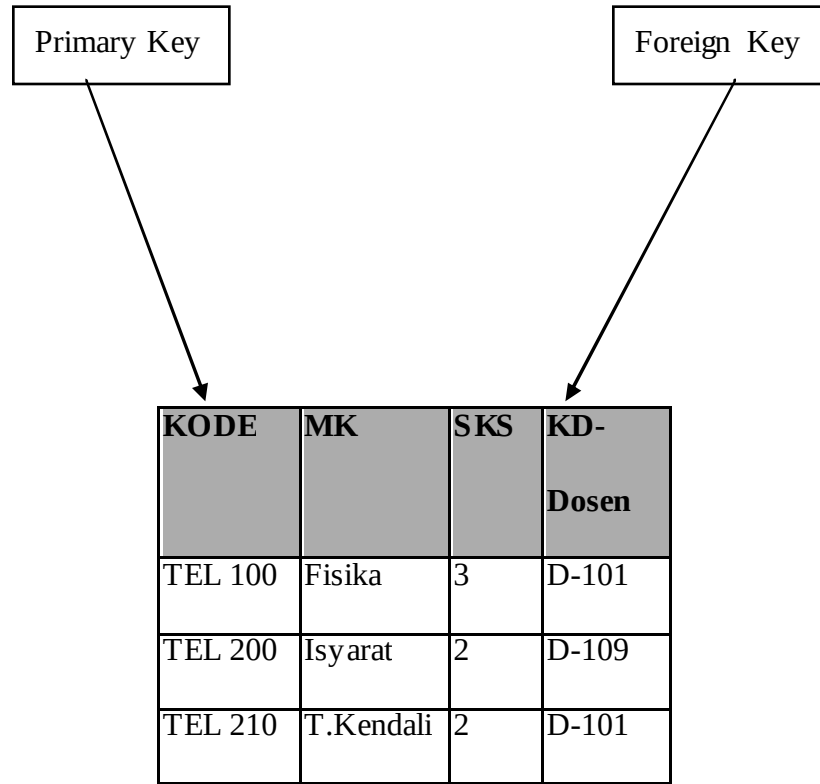
Primary key :Salah satu attrribut dari candidat key dapat dipilih menjadi primary key dengan 3 kriteria sbb :

- Key tersebut lebih natural untuk dijadikan acuan
- Key tersebut lebih sederhana
- Key tersebut cukup unqiue

Alternate key :Setiap atribut dari candidate key yang tidak terpilih sebagai primary key akan dinamakan alternate key. Pada contoh sebelumnya bila untuk primary key dipilih ID_Cus maka alternate key nya adalah No.of Pay

Foreign key : merupakan sembarang atribut yang menunjuk kepada primary key pada tabel lain.

- Akan terjadi pada suatu relasi yang memiliki kardinalitas one to many atau many to many

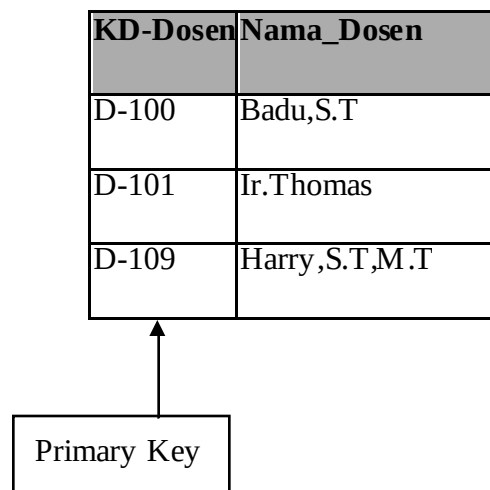


Primary Key

Foreign Key

KODE	MK	SKS	KD-Dosen
TEL 100	Fisika	3	D-101
TEL 200	Isyarat	2	D-109
TEL 210	T.Kendali	2	D-101

Table 2.2 tabel primary key dan foreign key



Primary Key

KD-Dosen	Nama_Dosen
D-100	Badu,S.T
D-101	Ir.Thomas
D-109	Harry,S.T,M.T

Table 2.3 tabel primary key

Tahap Perancangan Database

- Perancangan secara konseptual
 - o Diagram konteks
 - o DFD
 - o Model ER
- Perancangan secara logis
 - o Translasi model ER ke Model Relasional
- Perancangan secara fisik
 - o Penciptaan *database*, relasi, dan hal-hal terkait ke dalam bentuk fisik

2.2.6. MySQL

2.2.6.1. Pengenalan MySQL

MySQL adalah salah satu jenis *database server*, yang menggunakan *SQL* sebagai bahasa dasar untuk mengakses *database*. *Database* adalah sekumpulan tabel yang saling berhubungan satu sama lain. Sedangkan menurut C.J.Date, *database* adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara informasi dan membuat informasi tersedia saat dibutuhkan. MySQL termasuk *RDBMS* (*Relational Database Management System*). Untuk menambah, mengakses dan memproses data yang disimpan di komputer, diperlukan sistem manajemen *database* seperti MySQL. MySQL bersifat *free* atau gratis (tidak perlu membayar dalam menggunakannya). Namun MySQL terdiri dari dua lisensi, yaitu:

- Lisensi *free* (*free software/open source GNU General Public License*).

MySQL lisensi ini bebas digunakan, dimodifikasi *source* programnya dengan catatan harus dipublikasikan ke pemakai.

- Lisensi komersial (*non GPL*), pemakai harus membayar sejumlah biaya kepada *MySQL AB* sebagai pemegang hak cipta, sesuai dengan jenis layanan yang tersedia.

MySQL dapat digunakan pada berbagai *platform* sistem operasi. Khusus pada sistem operasi *Windows*, *MySQL* bersifat *shareware* (dikenai biaya setelah melakukan modifikasi dan digunakan untuk keperluan produksi). Keunggulan *MySQL* sebagai “*backend*” dalam mengolah *database* adalah:

- **Kecepatan.** Berdasarkan hasil pengujian, *MySQL* memiliki kecepatan yang paling baik dibanding *RDBMS* lainnya. Sebagai contoh, *MySQL 4.0* kinerja *query* naik sebesar 200% dari kinerja biasa.
- **Mudah digunakan.** Perintah dalam *MySQL* dan aturan-aturannya relatif mudah diingat dan diimplementasikan, karena *MySQL* menggunakan *SQL* (*Structured Query Language*) sebagai bahasa standar *database*.
- **Open source.** *MySQL* sudah menggunakan konsep *open source*, artinya siapapun dapat berkecimpung dalam mengembangkan *MySQL* dan hasil pengembangannya dipublikasikan kepada para pemakai.
- **Kapabilitas.** *MySQL* mampu memproses data yang tersimpan dalam *database* dengan jumlah 50 juta *record*, 60.000 tabel dan 5.000.000.000 jumlah baris, serta mampu memproses sebanyak 32 indek per-tabel.
- **Biaya murah.** Pemakai dapat menggunakan *MySQL* tanpa harus mengeluarkan biaya yang cukup mahal selama mengikuti konsep *open source/GNU Public*

License.

- **Keamanan.** *MySQL* menerapkan sistem keamanan dan hak akses secara bertingkat, termasuk dukungan dengan keamanan data secara pengacakan lapisan data. Adanya tingkatan *user* dan jenis akses yang beragam dan sistem pengacakan *password* (*encrypted password*).

- **Lintas platform.** *MySQL* dapat dijalankan pada beberapa sistem operasi, di antaranya yaitu *Linux*, *Windows*, *FreeBSD*, *Novell Netware*, *Sun Solaris* *SCO OpenUnix* dan *IBM's AIX*.

- **Minim “bug”.** Khususnya pada *MySQL* dengan keterangan “*recommended*”.

SQL (*Structure Query Language*) bukan suatu perangkat lunak atau bahasa pemrograman. Namun *SQL* adalah sebagai bahasa standar yang digunakan untuk mengolah *database*. Sedangkan perangkat lunak *MySQL* dalam mengolah *database*, menggunakan *SQL* sebagai bahasanya. *SQL* mempunyai kemampuan melakukan *query* dan manipulasi data, melakukan proses menciptakan *database* dan tabel, manajemen sekuritas hak akses, mendapatkan informasi, melakukan pemutakhiran data, dan sebagainya.

2.2.6.2 Administrasi Pada Database MySQL

Pertama, masuk ke *mysql* sebagai *user root*, kemudian pilih *database mysql* dengan menggunakan perintah ‘*use mysql;*’. Dalam *database mysql* terdapat beberapa tabel. Tabel tersebut dapat dilihat dengan mengetikkan perintah ‘*show tables;*’, maka akan tampil daftar tabel-tabel pada *database mysql* seperti berikut ini:

2.2.6.3 Tipe Tabel Pada MySQL

MySQL menyediakan berbagai tipe tabel yang dapat digunakan untuk membangun suatu *database*. Tipe-tipe tabel tersebut antara lain:

- *ISAM*

ISAM adalah tipe tabel dasar yang ada pada MySQL, yang merupakan tabel *non-transaksi*. Tabel ini sudah tidak banyak digunakan lagi dan sudah digantikan oleh tabel *MyISAM*. Karakteristik *ISAM* adalah sebagai berikut:

- o Panjang datanya bisa merupakan ukuran pasti/dinamis.
- o Tiap tabel dapat memuat 16 *index*
- o Maksimum *key length* adalah 256 *bytes*
- o Nilai data disimpan dalam *format* mesin
- o Tidak mendukung multi karakter per-tabel

- *MyISAM*

MyISAM merupakan pengganti dari *ISAM*, yang menangani tabel tunggal(*single table*). *MyISAM* memiliki karakteristik sebagai berikut:

- o Menyimpan nilai data dengan ukuran terkecil terlebih dahulu
- o Memfasilitasi *file* yang berukuran besar hingga 63 *bit file*
- o Maksimum jumlah per-tabel adalah 64
- o Setiap tabel disimpan dalam *file* yang terpisah
- o Maksimum *key-length* adalah 1000 *bytes*
- o Nilai “*NULL*” diijinkan pada kolom
- o Semua nilai *numerik* disimpan dengan ukuran terbesar terlebih dahulu untuk memungkinkan kompresi *index*

- o *File index* lebih kecil bila dibandingkan dengan *ISAM*, yang berarti bahwa *MyISAM* menggunakan lebih sedikit *system resources* daripada *ISAM*, tapi membutuhkan lebih banyak waktu *CPU* saat memasukkan data ke *index*
- o *MyISAM* secara otomatis meng-*update* kolom untuk *Insert/Update* yang menyebabkan fungsi *auto increment* untuk kolom tersebut lebih cepat
- o *MyISAM* menyediakan fasilitas untuk tipe *VarChar*, panjang awal disimpan dalam ukuran 2 *bytes*
- o Tabel dengan *VarChar* dapat menyimpan panjang yang sudah pasti/dinamis
- o Kolom *VarChar* dan *Char* dapat memiliki ukuran hingga 64KB

- *Merge Table*

Adalah kumpulan dari beberapa tabel *MyISAM* yang dapat digunakan sebagai satu kesatuan, dengan karakteristik sebagai berikut:

- o Dapat memasukkan data dari tabel yang berbeda
- o Memiliki kecepatan yang baik dan menggunakan ruang data yang lebih kecil
- o Lebih mudah dalam pencarian data
- o Lebih efisien dalam melakukan perbaikan, karena akan lebih mudah memperbaiki tabel-tabel tunggal yang dipetakan dalam *merge tabel* daripada memperbaiki 1 tabel yang besar

- *InnoDB*

- o Menyediakan fasilitas untuk dilakukannya transaksi yang aman
- o Tabelnya tidak dapat menyimpan lebih dari 1000 kolom
- o Pada beberapa sistem operasi yang lama, *file* data harus kurang dari 2GB.
- o Semua tabel disimpan dalam 1 kesatuan

- o Menyediakan fasilitas untuk *foreign key*
- o *InnoDB* didesain untuk mencapai fungsi yang maksimal saat memproses data yang besar
- *Memory(Heap) Table*
 - o Tabel ini menggunakan 100% *dynamic hashing* (saat memasukkan data). Tambahan ruang untuk kunci tidak dibutuhkan
 - o Baris yang dihapus diletakkan pada *linked list* dan akan digunakan kembali saat dilakukan masukan data baru ke tabel
 - o *Memory tables* memfasilitasi hingga 32 *index* per-tabel, 16 kolom per-*index*, dan maksimum *key length* 500 *bytes*
 - o Dapat menggunakan kunci yang tidak unik
 - o Menggunakan *format* panjang data yang sudah pasti
 - o Tidak menyediakan fungsi *auto increment*
 - o Tidak mengijinkan adanya nilai “*NULL*” pada kolom
 - o *Server* membutuhkan *memory* tambahan untuk mengatur seluruh tabel yang digunakan pada waktu yang bersamaan
- *BDB (Berkeley DB) tables*
 - o Dapat memuat hingga 31 *index* per-tabel, 16 kolom per-*index*, dan maksimum *key size* 1024 *bytes*
 - o *Primary key* lebih cepat daripada *index* yang lain, karena *primary key* disimpan bersama dengan baris data
 - o Informasi kunci menyita tempat lebih kecil dibandingkan dengan tabel *MyISAM*

- o Lambat bila membuka beberapa tabel secara bersamaan

- o Bila dalam kolom, yang bernilai “*NULL*” adalah *index* yang unik maka hanya 1 nilai “*NULL*” yang diijinkan

- *Archive table*

- o Hanya menyediakan fasilitas *insert* dan *select*(Tidak menyediakan fasilitas *delete, replace, update*)

- o Kompresi data langsung akan terjadi saat pertama kali dimasukkan.

- *Federated table*

- o Prosedur pembuatannya sangat mudah.

- o Merupakan tabel non-transaksi

- o Harus menggunakan *MySQL server*.

- o Tidak menyediakan fasilitas *alter table, drop table*.

Berdasarkan tipe-tipe tabel dan karakteristiknya masing-masing, maka dalam pembuatan *website e-commerce* ini dianjurkan menggunakan tipe table *InnoDB*.

2.2.6.4 Tipe data MySQL

Berikut ini akan diberikan tipe-tipe data yang didukung oleh MySQL yang terambil dari dokumentasi MySQL. Tipe - tipe data ini diberikan dalam bentuk yang siap dituliskan pada sintaks-sintaks MySQL, misalnya Create Table. Pada tipe-tipe data tersebut terdapat beberapa atribut yang memiliki arti sebagai berikut:

- M, menunjukkan lebar karakter maksimum. Nilai M maksimum adalah 255.

- D, menunjukkan jumlah angka di belakang koma. Nilai maksimum D adalah 30 tetapi dibatasi oleh nilai M, yaitu tidak boleh lebih besar daripada M-2.
- Atribut yang diberi tanda [dan] berarti pemakaiannya adalah optional.
- Jika atribut ZEROFILL disertakan, MySQL akan otomatis menambahkan atribut UNSIGNED.
- UNSIGNED adalah bilangan tanpa tanda di depannya (misalnya tanda negatif).

Inilah tipe-tipe data tersebut:

- TINYINT[(M)] [UNSIGNED] [ZEROFILL]

Integer yang sangat kecil jangkauan nilainya, yaitu -128 hingga 127. Jangkauan unsigned adalah 0 hingga 255.

- SMALLINT[(M)] [UNSIGNED] [ZEROFILL]

Integer yang kecil jangkauan nilainya, yaitu -32768 hingga 32767. Jangkauan unsigned adalah 0 hingga 65535.

- MEDIUMINT[(M)] [UNSIGNED] [ZEROFILL]

Integer tingkat menengah. Jangkauan nilainya adalah -8388608 hingga 8388607. Jangkauan unsigned adalah 0 hingga 16777215.

- INT[(M)] [UNSIGNED] [ZEROFILL]

Integer yang berukuran normal. Jangkauan nilainya adalah -2147483648 hingga 2147483647. Jangkauan unsigned adalah 0 hingga 4294967295.

- INTEGER[(M)] [UNSIGNED] [ZEROFILL]

Sama dengan INT.

- BIGINT[(M)] [UNSIGNED] [ZEROFILL]

Integer berukuran besar. Jangkauan nilainya adalah -9223372036854775808 hingga 9223372036854775807. Jangkauan unsigned adalah 0 hingga 18446744073709551615.

- FLOAT(precision) [ZEROFILL]

Bilangan floating-point. Tidak dapat bersifat unsigned. Nilai atribut precision adalah ≤ 24 untuk bilangan floating-point presisi tunggal dan di antara 25 dan 53 untuk bilangan floating-point presisi ganda.

- FLOAT[(M,D)] [ZEROFILL]

Bilangan floating-point presisi tunggal. Tidak dapat bersifat unsigned. Nilai yang diijinkan adalah -3.402823466E+38 hingga -1.175494351E-38 untuk nilai negatif, 0, and 1.175494351E-38 hingga 3.402823466E+38 untuk nilai positif.

- DOUBLE[(M,D)] [ZEROFILL]

Bilangan floating-point presisi ganda. Tidak dapat bersifat unsigned. Nilai yang diijinkan adalah $-1.7976931348623157E+308$ hingga $-2.2250738585072014E-308$ untuk nilai negatif, 0, dan $2.2250738585072014E-308$ hingga $1.7976931348623157E+308$ untuk nilai positif.

- `DOUBLE PRECISION[(M,D)] [ZEROFILL]` dan `REAL[(M,D)] [ZEROFILL]`

Keduanya sama dengan `DOUBLE`.

- `DECIMAL[(M[,D))]` `[ZEROFILL]`

Bilangan floating-point yang “unpacked”. Tidak dapat bersifat unsigned. Memiliki sifat mirip dengan `CHAR`. Kata “unpacked” berarti bilangan disimpan sebagai string, menggunakan satu karakter untuk setiap digitnya. Jangkauan nilai dari `DECIMAL` sama dengan `DOUBLE`, tetapi juga tergantung dai nilai atribut `M` dan `D` yang disertakan. Jika `D` tidak diisi akan dianggap 0. Jika `M` tidak diisi maka akan dianggap 10. Sejak MySQL 3.22 nilai `M` harus termasuk ruang yang ditempati oleh angka di belakang koma dan tanda + atau -.

- `NUMERIC(M,D)` `[ZEROFILL]`

Sama dengan `DECIMAL`.

- `DATE`

Sebuah tanggal. MySQL menampilkan tanggal dalam format 'YYYY-MM-DD'. Jangkauan nilainya adalah '1000-01-01' hingga '9999-12-31'.

- DATETIME

Sebuah kombinasi dari waktu (jam) dan tanggal. MySQL menampilkan waktu dan tanggal dalam format 'YYYY-MM-DD HH:MM:SS'. Jangkauan nilainya adalah '1000-01-01 00:00:00' hingga '9999-12-31 23:59:59'.

- TIMESTAMP[(M)]

Sebuah timestamp. Jangkauannya adalah dari '1970-01-01 00:00:00' hingga suatu waktu di tahun 2037. MySQL menampilkan tipe data TIMESTAMP dalam format YYYYMMDDHHMMSS, YYMMDDHHMMSS, YYYYMMDD, atau YYMMDD, tergantung dari nilai M, apakah 14 (atau tidak ditulis), 12, 8, atau 6.

- TIME

Tipe data waktu. Jangkauannya adalah '-838:59:59' hingga '838:59:59'. MySQL menampilkan TIME dalam format 'HH:MM:SS'.

- YEAR[(2|4)]

Angka tahun, dalam format 2- atau 4-digit (default adalah 4-digit). Nilai yang mungkin adalah 1901 hingga 2155, 0000 pada format 4-digit, dan 1970-2069 pada format 2-digit (70-69).

- CHAR(M) [BINARY]

String yang memiliki lebar tetap. Nilai M adalah dari 1 hingga 255 karakter. Jika ada sisa, maka sisa tersebut diisi dengan spasi (misalnya nilai M adalah 10, tapi data yang disimpan hanya memiliki 7 karakter, maka 3 karakter sisanya diisi dengan spasi). Spasi ini akan dihilangkan apabila data dipanggil. Nilai dari CHAR akan disortir dan diperbandingkan secara case-insensitive menurut default character set yang tersedia, kecuali bila atribut BINARY disertakan.

- VARCHAR(M) [BINARY]

String dengan lebar bervariasi. Nilai M adalah dari 1 hingga 255 karakter. Jika nilai M adalah 10 sedangkan data yang disimpan hanya terdiri dari 5 karakter, maka lebar data tersebut hanya 5 karakter saja, tidak ada tambahan spasi.

- TINYBLOB dan TINYTEXT

Sebuah BLOB (semacam catatan) atau TEXT dengan lebar maksimum 255 ($2^8 - 1$) karakter.

- BLOB dan TEXT

Sebuah BLOB atau TEXT dengan lebar maksimum 65535 ($2^{16} - 1$) karakter.

- MEDIUMBLOB dan MEDIUMTEXT

Sebuah BLOB atau TEXT dengan lebar maksimum 16777215 ($2^{24} - 1$) karakter.

- LONGBLOB dan LONGTEXT

Sebuah BLOB atau TEXT dengan lebar maksimum 4294967295 ($2^{32} - 1$) karakter.

- ENUM('value1','value2',...)

Sebuah enumerasi, yaitu objek string yang hanya dapat memiliki sebuah nilai, dipilih dari daftar nilai 'value1', 'value2', ..., NULL atau nilai special "" error. Sebuah ENUM maksimum dapat memiliki 65535 jenis nilai.

- SET('value1','value2',...)

Sebuah set, yaitu objek string yang dapat memiliki 0 nilai atau lebih, yang harus dipilih dari daftar nilai 'value1', 'value2', Sebuah SET maksimum dapat memiliki 64 anggota.

Nama field	Tipe data	Keterangan
Kode	Char(5)	Kode barang biasanya memiliki jumlah karakter tetap, dalam

		contoh ini adalah 5.
Nama	Varchar(20)	Nama barang kemungkinan besar memiliki jumlah karakter yang berbeda-beda, sehingga cocok menggunakan varchar.
Harga	Mediumint Unsigned	Harga tergantung dari jenis barangnya, pada contoh ini dianggap harga barang tertinggi di bawah 100 jutaan.
Supplier	Char(5)	Supplier biasanya juga dituliskan dalam kode tertentu yang jumlah karakternya tetap.
Tanggal	Date	Tanggal pembelian. Umumnya waktu pembelian tidak perlu disertakan, hanya tanggalnya saja.

Tabel 2.4 Tabel data MySql

2.2.7. PHP (PHP Hypertext Preprocessor)

2.2.7.1. Pengertian PHP

PHP merupakan bahasa *server-side* yang cukup handal, yang akan disatukan dengan *HTML* dan berada di *server*. Artinya, *sintaks* dan perintah yang diberikan akan sepenuhnya dijalankan di *server* sebelum dikirim ke komputer klien. Pada awal tahun 1995, Rasmus Lerdorf membuat produk bernama *PHP/FI* (*Personal Home Page/ Forms Interpreter*). Produk yang merupakan cikal bakal *PHP* ini ditulis menggunakan bahasa *C*, dan memiliki kemampuan untuk berkomunikasi dengan *database* serta membuat halaman dinamis.

Salah satu kelebihan *PHP* adalah kemampuan untuk dapat melakukan koneksi dengan berbagai *database*, seperti *MySQL*, *PostgreSQL*, dan *Access* sehingga sangat cocok jika digunakan untuk aplikasi *E-Commerce* seperti

shopping online. Selain itu *PHP* juga bersifat *open source*, untuk dapat menggunakannya kita tidak perlu membayar, cukup *download* pada situs *www.php.net*.

2.2.7.2 Variabel *PHP*

Variabel digunakan untuk menyimpan data yang nilainya dapat berubah-ubah. Dalam bahasa *PHP*, variabel dimulai dengan tanda “\$”. Aturan penulisan variabel antara lain sebagai berikut:

- Hanya ada 3 karakter yang dapat digunakan untuk nama variabel yaitu huruf, angka, dan garis bawah.
- Karakter pertama setelah tanda “\$” harus huruf atau garis bawah.
- Jika nama variabel lebih dari satu kata, tidak boleh ada tanda spasi di antara keduanya.

2.2.7.3 Keunggulan *PHP*

PHP Hypertext Preprocessor adalah skrip yang berjalan dalam *server side* yang ditambahkan dalam *HTML*. Skrip ini akan membuat suatu aplikasi dapat diintegrasikan ke dalam *HTML* sehingga suatu halaman *HTML* tidak lagi bersifat statis, namun menjadi bersifat dinamis. Sifat *server side* ini membuat pengerjaan skrip tersebut di kerjakan di *server* sedangkan yang di kirimkan kepada *browser* adalah hasil proses dari skrip tersebut yang sudah berbentuk *HTML*. Keunggulan dari sifatnya yang *server side* tersebut adalah:

- Tidak diperlukan adanya kompatibilitas *browser* atau harus menggunakan *browser* tertentu, karena *server*-lah yang akan mengerjakan skrip tersebut.

Hasil yang dikirimkan kembali ke *browser* biasanya dalam bentuk teks ataupun

gambar sehingga dapat dikenali oleh *browser* apa pun.

- Dapat memanfaatkan sumber-sumber aplikasi yang dimiliki oleh *server*, contoh: hubungan ke dalam *database*.
- Skrip asli tidak dapat dilihat sehingga keamanan lebih terjamin.

2.2.7.4. Koneksi *PHP* dan *MySQL*

Untuk menghubungkan bahasa pemrograman *PHP* dengan *MySQL* dibutuhkan beberapa perintah-perintah khusus, yang terdiri dari:

- a. Pembuatan koneksi antara *server* dari *MySQL* dengan *web server* tempat menyimpan halaman *web*. Dengan perintah:

```
<?
```

```
mysql_connect("nama server MySQL",login,password);
```

```
?>
```

- b. Setelah terbentuk koneksi maka selajutnya dilakukan pemilihan *database* yang akan digunakan dengan perintah:

```
<?
```

```
mysql_select_db("nama database");
```

```
?>
```

Jika belum pernah dibentuk suatu *database* maka dapat dibuat sebuah *database* dengan menggunakan perintah:

```
<?
```

```
mysql_create_db ("nama database baru");
```

```
?>
```

- c. Barulah kemudian dapat dilakukan perintah-perintah *SQL* yang lain seperti

select, update, delete, insert, dan perintah-perintah lainnya.

Contoh:

```
<?
```

```
Mysql_query("perintah query");
```

```
?>
```

Untuk memudahkan proses pengubahan, penambahan, penghapusan data pada *MySQL* dapat digunakan *software MySQL Front* yang dapat di jalan dibawah sistem operasi *windows*.

2.2.7.5 Apache Web Server

Agar *file-file MySQL* dapat dijalankan, maka diperlukan *web server* yang mampu memprosesnya. Untuk perancangan dan pembuatan tugas akhir ini, *web server* yang digunakan penulis adalah *Apache Web Server*. Dengan *Apache*, maka komputer akan dapat berfungsi sebagai *server*. Dan berfungsi sebagai *database server* dengan *MySQL*. Bahkan kini sudah ada aplikasi yang mengintegrasikan 3 *software PHP, Apache* dan *MySQL*, yaitu *PHPTriad* yang dapat didownload secara gratis di <http://www.download.com>. Atau dapat mendownload *Apache* pada alamat <http://httpd.apache.org>.

2.2.8 UML

Menurut Rama dan Jones (2006,p60), *Unified Modeling Language (UML), a language used for specifying, visualising, constructing, and documenting an information system.*

Dapat disimpulkan, UML adalah salah satu dari beberapa teknik yang dapat digunakan dalam mendokumentasi proses bisnis. UML dapat digunakan untuk memahami berbagai dokumen sistem informasi.

2.2.9 Rich picture

Menurut Mathiassen, dkk (2000, p335), *Rich Picture* adalah sebuah gambaran dari, obyek, proses, struktur, dan masalah dalam sistem dan *application domain*. Digunakan selama pemilihan sistem untuk mengekspresikan keseluruhan persepsi dari tugas menghadapi proyek pengembangan sistem. *Rich picture* tidak digambarkan dengan notasi khusus. Bagaimanapun *rich picture* yang dibuat harus sesuai dengan proyek dan menjelaskan aspek yang akan dideskripsikan.

2.2.10. Use Case

Menurut Jones dan Ramo (2006, p267) *usecase diagram is a list of use cases that occur in a application and that indicate the actor responsible for each use case.*

Use case Diagram adalah daftar dari use case yang terjadi dalam aplikasi dan yang mengindikasikan tanggung jawab aktor dari setiap use case.

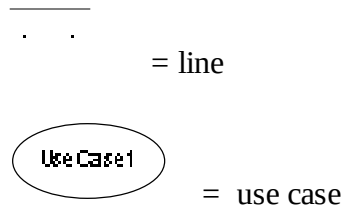
Macam – macam symbol use case :



= sistem boundary (untuk membatasi ruang lingkup sistem)

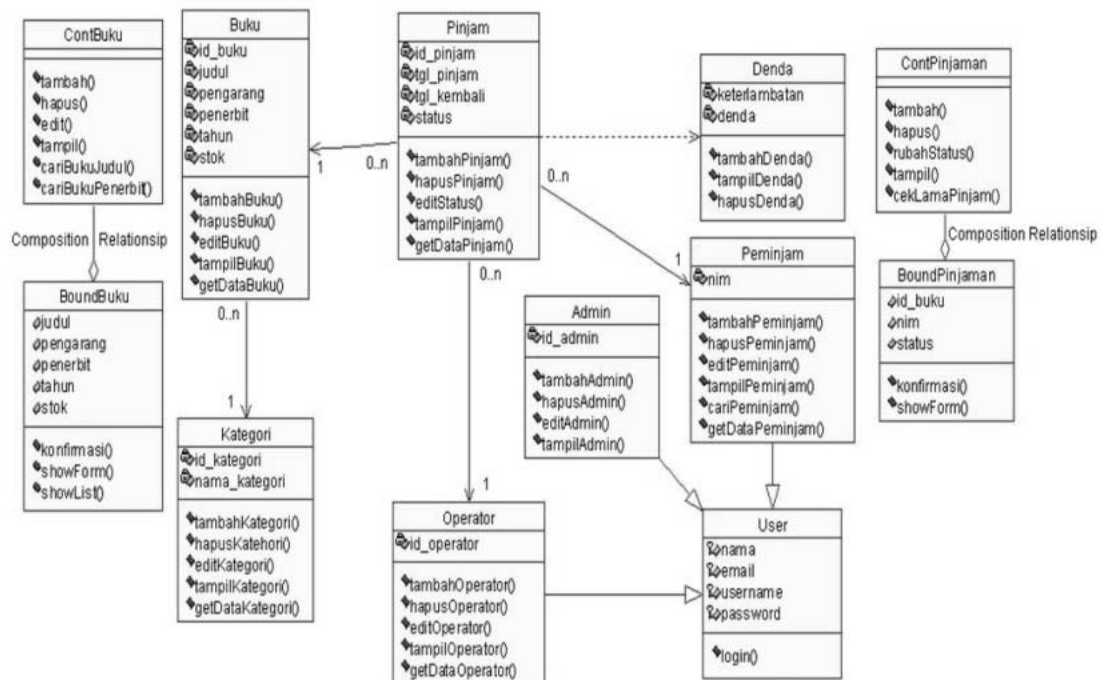


= actor



2.2.11 Class Diagram

Class diagram menunjukkan class yang terdapat dalam perangkat lunak dan bagaimana mereka saling berhubungan. Class diagram mendeskripsikan jenis – jenis objek dalam sistem dan berbagai hubungan statis yang terdapat diantara mereka. Class diagram juga menunjukkan property dan sebuah kelas dan batasan – batasan yang terdapat dalam hubungan – hubungan objek tersebut (Fowler, 2005, p53).



Gambar 2.1 contoh class diagram

2.2.11.1 Pengertian Class

Menurut Mathiassen, Madsen, Nielsen dan Stage (2002,p53), *Class* : A description of a collection of objects sharing structure, behaviour pattern, and attributes.

Biasanya setelah kita mengidentifikasi semua *object* dan *event* yang ada, maka kita akan menyertakannya dalam model *problem domain*. Untuk mempermudah pengembangan *problem domain* tersebut, maka kita kelompokkan beberapa *objek* dan *event* tersebut ke dalam beberapa *class*.

2.2.11.2 Pengertian Attribute

Menurut Mathiassen, Madsen, Nielsen dan srage (2002,p89), *Attribute* : A description property of a class or event.

Menurut Rama dan Jones (2006,p155), *Attribute*. The smallest units of data that can have meaning to user. The columns in a relational database that are equivalent to fields in a file.

Saat mendesain *problem domain* harus ada formulasi kebutuhan data yang akan disimpan oleh sistem, data-data tersebut harus dapat diakses melalui *functions* dan *interfaces*, maka untuk menspesifikasinya digunakanlah *attribute*. Dapat disimpulkan bahwa, dalam pendekatan *analisa object oriented*, yang dimaksud *attribute* adalah spesifikasi dari definisi *class*.

2.2.11.3 Pengertian Behaviour

Menurut Mathiassen, Madsen, Nielsen dan srage (2002,p90), *in the class activity behaviour simply as the unordered set of event that involve an object*.

Menurut Mathiassen, Madsen, Nielsen dan srage (2002,p90), *in the*

behaviour activity, we describe behaviour more precisely by adding the relative timing of events.

Secara lebih mudah dapat dikatakan bahwa *behaviour* adalah event-event yang memungkinkan dilakukan oleh semua *object* dalam *class*.

2.2.11.4 Pengertian Event

Menurut Mathiassen, Madsen, Nielsen dan srage (2002,p54), *event is an instantaneous incident one or more object.*

Menurut Rama dan Jones (2006,p4), *event are activities tht happen at a particular point in time.*

Sebagai contoh, misal *event customer* menempatkan *order* atau mengirimkan pesanan barang atau mencetak laporan.

Dengan kata lain, *event* adalah kegiatan atau kejadian yang merupakan bagian proses dari suatu sistem.

Simbol yang digunakan dalam class diagram yaitu :

1. Kelas

Kelas digambarkan dengan sebuah kotak yang terbagi atas tiga bagian, yaitu :

- a. Nama Kelas (bagian atas)
- b. Atribut kelas (bagian tengah)
- c. Operasi kelas (bagian bawah)

2. Hubungan

Di dalam clsss diagram terdapat 3 jenis hubungan yang umum digunakan, yaitu :

a. Asosiasi

Asosiasi digambarkan dengan sebuah garis lurus yang menghubungkan dua buah kelas, berawal dari kelas sumber menuju kelas yang dituju.

b. Generalisasi

Generalisasi digambarkan dengan sebuah garis yang biasanya digambarkan secara vertikal antara sebuah kelas dengan subkelasnya. Kelas sumber ditunjukkan menggunakan gambar segitiga yang menunjuk ke arah kelas tersebut. Di dalam sudut pandang software, yang dimaksud dengan generalisasi adalah pewarisan (inheritance).

c. Dependensi

Dependensi ada diantara 2 elemen yang saling berkaitan, dimana perubahan definisi dari elemen yang satu (supplier) juga menyebabkan perubahan definisi di elemen yang lainnya (client). Kelas client ditunjukkan dengan menggunakan segitiga yang menunjuk ke arah kelas tersebut.

2.2.12 Activity Diagram

Menurut Booch(1999,p260) Activity diagram adalah sebuah proyeksi yang bersifat dasar dari elemen – elemen yang ditemukan dalam sebuah grafik aktivitas, suatu kasus yang khusus dari state machine dalam semua atau kebanyakan transisi dipercepat dengan penyelesaian dari aktivitas – aktivitas dalam state number.

Suatu activity diagram adalah sejenis state machine, maka activity diagram berisi semua karakteristik dari penerapan state machine. Ini berarti activity diagram berisi state – state, cabang – cabang, fork (percabangan), join(gabungan) yang sederhana dan komposit.

- b. Action state adalah suatu tipe dari state yang mempresentasikan aktivitas yang lengkap
- c. Transisi dalam sebuah state terjadi apabila aksi dalam sebuah action state telah lengkap
- d. Transisi join dilakukan untuk mengidentifikasi aktivitas yang terjadi bersamaan yang harus dilengkapi sebelum aktivitas berikutnya terjadi.
- e. Transisi yang di double click ke labelnya dengan kondisi penjagaan dan ekspresi – ekspresi suatu aksi
- f. Ketika, suatu keputusan mendominasi ke yang lain maka gunakan bentuk berlian (belah ketupat) untuk mengidentifikasi yang kedua yaitu nested decision.
- g. Untuk mengidentifikasi aktivitas yang dapat dilakukan secara paralel maka gunakan transaksi fork.
- h. Final state mempresentasikan penyelesaian dari aktivitas dalam situasi yang direpresentasikan diagram.
- i. Swimlane berguna untuk merancang aktivitas secara prosedural dari kontrol diantara objek – objek (orang, organisasi, dan instansi lain yang bertanggung jawab) yang melakukan aksi – aksi.