BAB II

TINJAUAN PUSTAKA

2.1 Tinjauan Perusahaan

Berikut ini dijelaskan tentang tinjauan perusahaan yang meliputi riwayat perusahaan, struktur organisasi, beserta deskripsi jabatan yang ada pada Grenouille Shop Yogyakarta.

2.1.1 Riwayat Perusahaan

Grenouile Shop Yogakarta yang sekarang berdiri sebelumnya mempunyai latar belakang pendiriannya, yang mana *outlet* tersebut diawali dengan keberhasilan Ferry Sugiarto sebagai usahawan muda dari Yogyakarta. Ferry memulai usahanya dengan bisnis pakaian daerah dengan ukuran toko 3 x 4 meter, hanya saja toko tersebut sepi pengunjung, sehingga mendorong Ferry untuk mencoba bisnis baru di kawasan Sleman, yang kemudian menjadi *trend* di sekitar Sleman.

Menyusul bisnis yang semakin naik, maka pada tahun 1999, ia telah membuka usaha untuk pertama kalinya. Ia sempat dikenal sebagai pelopor cikal bakalnya *Factory Outlet*, dimana ia mempopulerkan barang sisa ekspor. Di tahun ini pula *Factory Outlet* pun menjamur di beberapa daerah, sampai ke Magelang dan Surabaya, untuk menghindari kejenuhan pasar, beliau terus berusaha menciptakan inovasi. Salah satu usahanya yaitu dengan membuka *factory outlet* bergaya *galery* dengan nama Grenouille Shop, yang berlokasi di Jl. Dr.Wahidin, Pringgolayan,

Yogyakarta. Grenouille Shop adalah sebuah *factory outlet* menitikberatkan pada kebutuhan wanita, dimana sebagian besar produknya adalah produk untuk wanita. Namun produk untuk priapun disediakan di *factory outlet* ini.

Pada *factory outlet* Grenouille Shop anda akan menjumpai toko dengan suasana yang berbeda dengan *factory outlet – factory outlet* lainnya, dimana ruangannya memiliki konsep – konsep yang berbeda dan unik. Seperti contohnya terdapat ruangan *Ekslusif* yang menyediakan pakaian formal seperti setelan jas, kemeja – kemeja kantor, dasi, dan sebagainya. Di sisi ruang lain ada ruangan yang berkonsep *Casual*, dimana ruangan ini menyediakan pakaian-pakaian bagi mereka pria yang gemar dengan model pakaian *casual*, di ruangan lainnya terdapat koleksi celana jeans yang beraneka ragam. Tersedia juga parfum, kaca mata dan berbagai macam aksesoris – aksesoris lainnya.

Menjamurnya *factory outlet* memberikan dampak positif bagi konsumen, sehingga konsumen semakin banyak memiliki pilihan. Untuk menghadapi persaingan yang semakin ketat setiap kalinya, Grenouille Shop selalu melakukan evaluasi untuk menentukan strategi baru dan berusaha untuk selalu menjadi yang terdepan. Sesuai dengan prinsip bahwa tak ada yang abadi di dunia ini selain perubahan, maka pihak manajemenpun berusaha untuk selalu siap mengikuti perubahan dan memberikan yang terbaik untuk konsumennya.

Visi dari *Grenouille Shop* ini adalah menjadi *fatory Outlet* yang unik. Dimana *Grenouille Shop* dapat memanjakan setiap konsumen yang gemar berbelanja, sedangkan misi dari *Grenouille Shop* ini adalah membuat setiap konsumennya merasa nyaman berada di *Grenouille Shop* dan terpenuhi kebutuhannya serta menemukan suasana yang berbeda dengan *factory outlet* lainnya .

2.1.2 Struktur Organisasi *Grenouille Shop* Yogyakarta

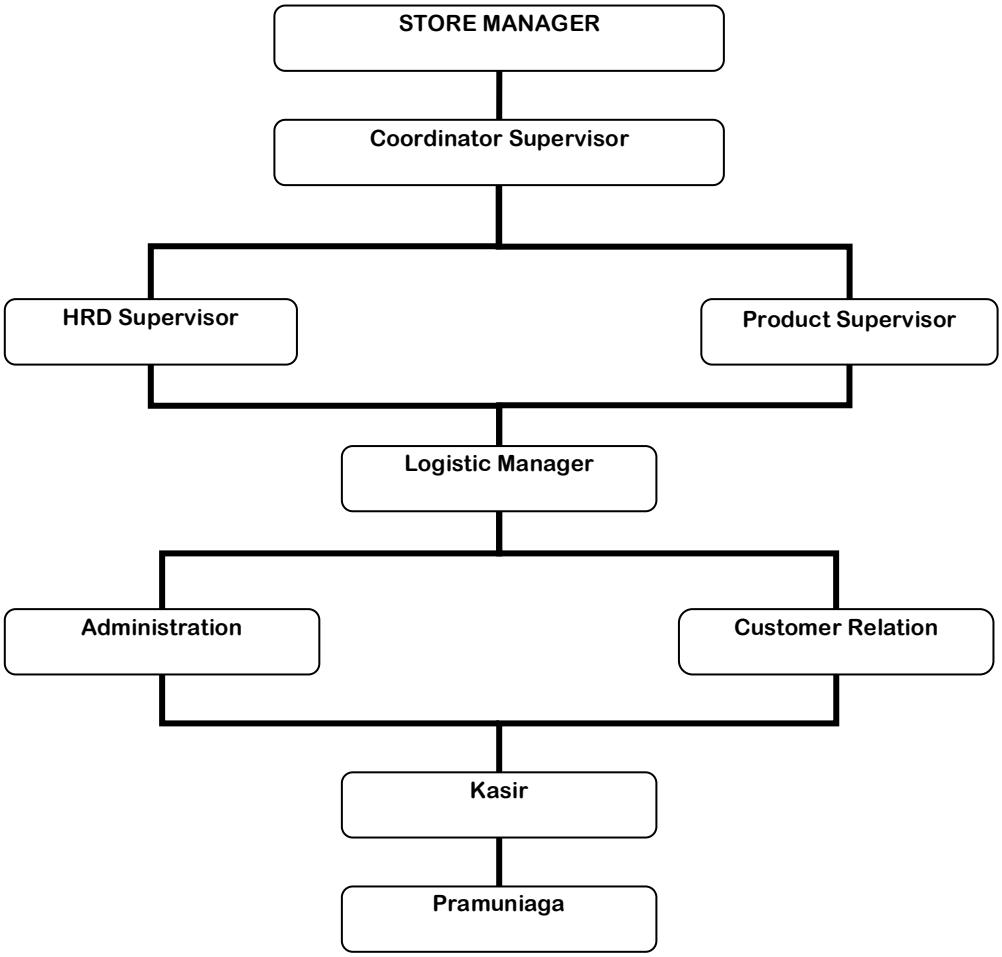
Struktur organisasi merupakan kerangka kerja yang menggambarkan hubungan kerja, wewenang dan tanggung jawab setiap tingkat yang ada dalam organisasi tersebut, untuk melaksanakan kegiatan ke arah tercapainya tujuan dari organisasi yang telah ditetapkan, sehingga tercapainya kerjasama dan koordinasi usaha diantara setiap unit organisasi dalam mengambil tindakan dan mencapai tujuan. Struktur organisasi yang baik merupakan sesuatu yang sangat penting bagi perusahaan, karena dengan struktur orgnisasi yang baik dan tepat dapat membantu kelancaran jalannya usaha yang baik dan teratur. Dengan organisasi yang baik dan tepat setiap karyawan akan ditempatkan pada posisi yang sesuai dengan minat dan keahlian dari masing – masing staf dan karyawannya. Begitu juga yang dilakukan secara selektif yaitu melihat kemampuan, bakat, dan minat dari karyawannya.

Agar tercapai efisiensi dan efektivitas bagi setiap karyawan dalam bekerja, perusahaan perlu menyusun dan menetapkan bagan organisasi yang disertai uraian tugas dan tanggung jawab di lingkungan perusahaan. Hal ini dilakukan untuk

menghindari *bias* atau kerancuan dalam pelaksanaan tugas, wewenang dan tanggung jawab masing-masing karyawan.

Struktur organisasi Grenouille Shop Yogyakarta dipimpin oleh *Store Manager*, *Store Manager* memberikan pelimpahan wewenang pada *Coordinator Supervisor*, *Coordinator Supervisor* memberikan wewenang lagi kepada *HRD Supervisor* dan *Product Supervisor*, *HRD Supervisor* dan *Product Supervisor* bekerja bersamaan serta saling memberikan saran-saran dan kritik yang membantu dalam pekerjaannya masing-masing, *HRD Supervisor* dan *Product Supervisor* memberikan pelimpahan wewenang kepada Kepala Gudang yang juga memberikan pelimpahan wewenang kepada *Administration* dan *Customer*. Kasir mendapatkan pelimpahan wewenang dari *Administration* dan *Customer Relation*. Pramuniaga berada di urutan paling bawah dan harus bertanggung jawab pada atasannya dalam melakukan pekerjaannya, tetapi tetap semua pihak harus bertanggung jawab pada pimpinan perusahaan atau *Store Manager.*

Adapun struktur organisasi Grenouille Shop Yogyakarta dapat digambarkan pada gambar 2.1 yaitu sebagai berikut:



Gambar 2.1 Struktur Organisasi *Grenouille Shop* Yogyakarta

Sumber: Grenouille Shop, Agustus 2010

Berdasarkan bagan struktur organisasi Grenouille Shop Yogyakarta pada gambar 2.1 dapat disimpulkan bahwa bentuk organisasi ini adalah lini dan staf. Organisasi lini dan staf ini pada umumnya memiliki asas kesatuan komando yang tetap dipertahankan. Pelimpahan wewenang berlangsung secara vertikal dan sepenuhnya dari pimpinan tertinggi kepada unit di bawahnya.

Dalam membantu kelancaran kerja pimpinan, ia mendapat bantuan dari para staf. Tugas dari para staf hanya untuk memberikan bantuan, saran – saran dan pelayanan kepada pimpinan sebagai bahan pertimbangan dalam mengambil kebijaksanaan/ keputusan. Garis wewenang tetap berada pada pimpinan, sedangkan staf hanya memiliki wewenang staf saja. Dipakai atau tidaknya saran – saran, nasehat-nasehat dari para staf, sepenuhnya tergantung pada pimpinan sendiri.

2.1.3 Deskripsi Jabatan *Grenouille Shop* Yogyakarta

Adapun tugas dari masing-masing departemen dari struktur organisasi diatas diantaranya:

1. ***Store Manager***
    a. menyusun dan menetapkan rencana kerja secara periodik dalam upaya pencapaian target,
    b. menyusun strategi penjualan,
    c. menerima semua laporan kegiatan operasional,
    d. melakukan *briefing* kepada *coordinator supervisor*,
    e. melakukan penilaiaian berkala kinerja *supervisor*.
2. ***Coordinator Supervisor***
    a. mengkoordinir semua kegiatan operasional yang berada di bawah tanggungjawabnya, seperti order, stok penjualan, *display*, penjualan, kebersihan, kenyamanan, keamanan serta kepuasan konsumen,

b. melakukan *briefing* kepada *supervisor* .

3. HRD *Supervisor*

a. mengawasi kehadiran serta keberadaan semua karyawan yang bertugas,

b. melakukan *briefing* kepada *coordinator supervisor* dan timnya ,

c. melakukan penilaian berkala kinerja semua karyawan,

d. membuat laporan penjualan mingguan dan bulanan.

4. *Product Supervisor*

a. mengawasi ketersediaan dan kelengkapan barang baik setiap hari maupun even khusus,

b. memastikan sarana dan prasarana yang ada di lapangan masih berfungsi dengan baik,

c. melakukan *survey competitor* dan melakukan analisa perbaikan,

d. membuat laporan penjualan dan omzet migguan dan bulanan.

5. *Logistic Manager*

a. bertanggung jawab terhadap kelengkapan dan kesediaan barang,

b. melakukan perputaran produk jual yang ada di dalam toko.

6. *Administration*

a. mengeluarkan surat edaran resmi dari toko,

b. mencatat absen kehadiran karyawan,

c. membuat laporan pertanggungjawaban absen kehadiran karyawan setiap bulannya.

7. *Customer Relation*

a. bertanggung jawab terhadap *display*,

b. bertanggung jawab terhadap kebersihan, media *display* dan *display* pendukungnya,

c. membina hubungan baik dengan pelanggan,

8. Kasir

a. menerima pembayaran dari produk yang terjual,

b. membuat laporan penjualan stiap hari.

9. Pramuniaga

a. melayani *customer*,

b. menjaga dan membina hubungan baik dengan *customer*,

c. menjaga kebersihan *counter*.

2.2 Landasan Teori

2.2.1 Internet

Internet adalah rangkaian atau jaringan sejumlah komputer yang saling berhubungan. Internet berasal dari kata *interconnected-networking*. Internet merupakan jaringan global yang menghubungkan suatu jaringan (*network*) dengan jaringan lainnya di seluruh dunia. Media yang menghubungkan bisa berupa kabel, kanal satelit maupun frekuensi radio.

Jaringan internet bekerja berdasarkan suatu protokol (aturan). TCP/IP yaitu *Transmission Control Protocol Internet Protocol* adalah protokol standar yang

digunakan untuk menghubungkan jaringan-jaringan di dalam internet sehingga data dapat dikirim dari satu komputer ke komputer lainnya. Setiap komputer diberikan suatu nomor unik yang disebut dengan alamat IP.

Sejarah internet dimulai dari ARPANet, yaitu sebuah proyek Departemen Pertahanan Amerika Serikat. Pada tahun 1969 dilakukan sebuah riset yaitu bagaimana cara menghubungkan suatu komputer dengan komputer lainnya atau membentuk suatu jaringan. Di tahun 1970 mereka berhasil menghubungkan lebih dari 10 komputer yang membentuk jaringan. Kemudian tahun 1973 jaringan ARPANet mulai dikembangkan di luar Amerika Serikat. Sejarah internet berlanjut ketika komputer di University College di London ikut bergabung dengan jaringan ARPANet. Di tahun-tahun selanjutnya jaringan ini semakin berkembang.

Pada tahun 1982, karena banyaknya komputer yang bergabung maka dibutuhkan sebuah protokol resmi yang menghubungkan semua komputer dan jaringan dan dibentuklah TCP/IP. Di tahun 1984 kemudian diperkenalkan sistem penamaan domain yang dikenal dengan DNS (Domain Name System). Internet kemudian tumbuh secara pesat di tahun 1990-an.

2.2.2 *Web Server*

Web Server adalah sebuah perangkat lunak server yang berfungsi menerima permintaan HTTP atau HTTPS dari klien yang dikenal dengan web browser dan mengirimkan kembali hasilnya dalam bentuk halaman-halaman web yang umumnya

berbentuk dokumen HTML. Web server menunggu permintaan dari client yang menggunakan browser seperti Netscape Navigator, Internet Explorer, Modzilla, dan program browser lainnya. Jika ada permintaan dari browser, maka *web server* akan memproses permintaan itu kemudian memberikan hasil prosesnya berupa data yang diinginkan kembali ke *browser*. Data ini mempunyai format yang standar, disebut dengan format SGML (*standar general markup language*). Data yang berupa format ini kemudian akan ditampilkan oleh browser sesuai dengan kemampuan browser tersebut. Contohnya, bila data yang dikirim berupa gambar, browser yang hanya mampu menampilkan teks (misalnya *lynx*) tidak akan mampu menampilkan gambar tersebut, dan jika ada akan menampilkan alternatifnya saja.

Untuk berkomunikasi dengan *client*-nya (*web browser*) *web server* mempunyai protokol sendiri, yaitu HTTP (*hypertext transfer protocol*). Dengan protokol ini, komunikasi antar *web server* dengan client-nya dapat saling dimengerti dan lebih mudah. Seperti telah dijelaskan diatas, format data pada *world wide web* adalah SGML. Tapi para pengguna internet saat ini lebih banyak menggunakan format HTML (*hypertext markup language*) karena penggunaannya lebih sederhana dan mudah dipelajari. Kata *HyperText* mempunyai arti bahwa seorang pengguna internet dengan *web browser*nya dapat membuka dan membaca dokumen-dokumen yang ada dalam komputernya atau bahkan yang jauh dari tempatnya sekalipun. Hal ini memberikan cita rasa dari suatu proses yang tridimensional, artinya pengguna internet dapat membaca dari satu dokumen ke dokumen yang lain hanya dengan mengklik beberapa bagian dari halaman-halaman dokumen (web) itu.

Proses yang dimulai dari permintaan *web client* (browser), diterima web server, diproses, dan dikembalikan hasil prosesnya oleh *web server* ke *web client* lagi dilakukan secara transparan. Setiap orang dapat dengan mudah mengetahui apa yang terjadi pada tiap-tiap proses. Secara garis besarnya *web server* hanya memproses semua masukan yang diperolehnya dari *web client*nya.

2.2.3 *Web browser*

Web browser adalah suatu program komputer yang menyediakan fasilitas untuk membaca halaman web di suatu komputer. Contohnya adalah seperti Microsoft Internet Explorer, Mozilla Firefox, Flock, dan lain sebagainya.

2.2.4 *Website* (Situs Web)

Secara terminologi, *website* adalah kumpulan dari halaman-halaman situs, yang biasanya terangkum dalam sebuah domain atau subdomain, yang tempatnya berada di dalam *World Wide Web* (WWW) di Internet. Sebuah halaman *web* adalah dokumen yang ditulis dalam format HTML (*Hyper Text Markup Language*), yang hampir selalu bisa diakses melalui HTTP, yaitu protokol yang menyampaikan informasi dari *server website* untuk ditampilkan kepada para pemakai melalui *web browser*. Semua publikasi dari *website* - *website* tersebut dapat membentuk sebuah jaringan informasi yang sangat besar.

Halaman-halaman dari website akan bisa diakses melalui sebuah URL yang biasa disebut Homepage. URL ini mengatur halaman-halaman situs untuk menjadi sebuah hirarki, meskipun, *hyperlink-hyperlink* yang ada di halaman tersebut mengatur

para pembaca dan memberitahu mereka sususan keseluruhan dan bagaimana arus informasi ini berjalan. Beberapa *website* membutuhkan subskripsi (data masukan) agar para *user* bisa mengakses sebagian atau keseluruhan isi *website* tersebut. Contohnya, ada beberapa situs-situs bisnis, situs-situs *e-mail* gratisan, yang membutuhkan subkripsi agar kita bisa mengakses situs tersebut.

Sebuah *website* statik, adalah salah satu bentuk *website* yang isinya tidak dimaksudkan untuk di *update* secara berkala, dan biasanya di *maintain* secara *manual* oleh beberapa orang yang menggunakan *software editor*. Terdapat tiga tipe kategori *software editor* yang biasa dipakai untuk tujuan *maintaining* ini, mereka adalah :

1. Elemen 1 Penyunting teks. Contohnya adalah Notepad atau TextEdit, dimana HTML diubah didalam *program editor* tersebut.
2. Elemen 2 WYSIWYG editor. Contohnya Microsoft Frontpage dan Macromedia Dreamweaver, dimana situs di *edit* menggunakan GUI (*Graphical User Interface*) dan format HTML ini secara otomatis di *generate* oleh *editor* ini.
3. Elemen 3 *Editor* yang sudah memiliki *template*, contohnya Rapidweaver dan iWeb, dimana, *editor* ini membolehkan *user* untuk membuat dan meng-*update website*-nya langsung ke *server web* secara cepat, tanpa harus mengetahui apapun tentang HTML. Mereka dapat memilih *template* yang sesuai dengan keinginan mereka, menambah

gambar atau obyek, mengisinya dengan tulisan, dan dengan sekejap mereka sudah dapat membuat situs *web* tanpa harus melihat sama sekali kode-kode HTML.

Sebuah *website dynamic* adalah *website* yang secara berkala, informasi didalamnya berubah, atau *website* ini bisa berhubungan dengan *user* dengan berbagai macam cara atau metode (HTTP *cookies* atau *Variabel Database*, sejarah kunjungan, variabel sesi dan lain-lain) bisa juga dengan cara interaksi langsung menggunakan *form* dan pergerakan *mouse*. Ketika *web server* menerima permintaan dari *user* untuk memberikan halaman tertentu, maka halaman tersebut akan secara otomatis di ambil dari media penyimpanan sebagai respon dari permintaan yang diminta oleh *user*. Sebuah situs dapat menampilkan dialog yang sedang berlangsung diantara dua *user*, memantau perubahan situasi, atau menyediakan informasi yang berkaitan dengan sang *user*.

Terdapat banyak jenis sistem *software* yang dapat dipakai untuk meng-*generate Dynamic Web System* dan Situs *Dynamic*, beberapa diantaranya adalah ColdFusion (CFM), Active Server Pages (ASP), Java Server Pages (JSP) dan PHP. Situs juga bisa termasuk didalamnya berisi informasi yang diambil dari satu atau lebih database atau bisa juga menggunakan teknologi berbasis XML, contohnya adalah RSS. Isi situs yang statis juga secara periodik di *generate*, atau, apabila ada

keadaan dimana dia butuh untuk dikembalikan kepada keadaan semula, maka dia akan di *generate*, hal ini untuk menghindari kinerjanya supaya tetap terjaga.

Plugin tersedia untuk menambah banyaknya *feature* dan kemampuan dari *web browser*, dimana, *plugin* ini dipakai untuk membuka *content* yang biasanya berupa cuplikan dari gambar bergerak (*active content*) contohnya adalah Flash, Shockwave atau applets yang ditulis dalam bahasa JAVA. *Dynamic* HTML juga disediakan untuk pengguna supaya bisa secara interaktif dan *realtime*, meng-*update* di *web page* tersebut, biasanya perubahan yang dilakukan mereka memakai DOM dan Javascript yang sudah tersedia pada semua *Web Browser* sekarang ini.

Seperti yang tertulis di atas, di luar sana ada beberapa perbedaan dalam penulisan dari terminologi *website*. Walaupun *website* sudah secara umum dipakai, namun untuk Associated Press Stylebook, Reuters, Microsoft, Academia, dan kamus-kamus yang ada, penulisan yang mereka pakai adalah dengan menggunakan 2 kata, yaitu *Web site*. Hal ini karena ¨*Web*¨ bukanlah terminilogi umum, namun kependekan dari *World Wide Web*.

2.2.5 *Electronic Commerce (E-Commerce)*

Electronic Commerce (EC) adalah suatu konsep yang menjelaskan proses pembelian, penjualan dan pertukaran produk, servis dan informasi melalui jaringan komputer yaitu *internet* (Turban, 2002). Menurut Kalakota (1999) *e-commerce* dapat dilihat dari sudut pandang :

a. Sudut pandang komunikasi, *e-commerce* merupakan pengiriman barang, servis, informasi, atau pembayaran melalui jaringan komputer.
b. Sudut pandang bisnis proses, *e-commerce* adalah aplikasi teknologi yang dapat melakukan transaksi bisnis dan arus kerja yang otomatis.
c. Sudut pandang servis, *e-commerce* merupakan peralatan yang dapat memenuhi keinginan perusahaan, konsumen dan managemen untuk memotong biaya servis selama pengembangan kualitas barang dan peningkatan kecepatan layanan pengiriman.
d. Sudut pandang *online*, *e-commerce* menyediakan kemampuan untuk membeli dan menjual produk dan informasi melalui *internet* dan layanan *online* lainnya.

Secara umum ada tiga tipe *e-commerce* yaitu:

a. B2C (*Business to Customer*), dalam B2C transaksi *online* dibuat antara bisnis dengan konsumen secara individual. Contoh : pesanan buku secara *online*, pembelian tiket pesawat terbang.
b. B2B (*Business to Business*), dalam B2B bisnis membuat transaksi *online* dengan bisnis lain. Contoh : Layanan *online* atau pembelian bahan bakar.
c. B2E (*Business to Employee*), dalam B2E informasi dan servis dibuat secara *online* untuk para pekerja. Contoh : Pelatihan *online* dan perbankan *online*.

E-commerce dapat dibedakan menjadi dua bagian yaitu *pure e-commerce* dan *partial e-commerce*. *Pure e-commerce* menerangkan semua bagian dalam *e-commerce* yang menggunakan dimensi *digital* baik dalam produk, proses dan pengiriman. *Pure e-commerce* sulit untuk dilakukan karena apabila produk dalam bentuk *hardgoods* maka pengiriman tidak dapat dilakukan melalui dimensi digital. Sedangkan yang kedua adalah *partial e-commerce*, menjelaskan bahwa *e-commerce* menggunakan semua kemungkinan untuk menggabungkan dimensi *digital* dan *physical*. Dalam penelitian ini akan digunakan metode *partial e-commerce*, dimana produk yang dijual berupa produk *hard goods* sehingga pengiriman dilakukan melalui jasa pengiriman sedangkan pembayaran dilakukan secara *digital* melalui pembayaran *online.*

2.2.6 *Electronic Retailing (E-Retailing)*

E-Retailing termasuk dalam jenis usaha E-COMMERCE yang memperdagangkan barang secara langsung kepada konsumen (*direct selling*). Bila digolongkan maka *E-Retailing* adalah golongan *bisnis to customer* (B2C) yang berarti golongan usaha dagang langsung atau secara eceran/*retail* (Turban, 2002). Dalam perdagangan B2C secara *online*, terdapat dua jenis barang yang ditawarkan, yaitu : *Hard goods* adalah barang yang berwujud nyata. Bila dibeli maka dikirimkan pada pelanggan melalui jasa pengiriman. *Soft goods* sering juga disebut sebagai *digital goods* seperti berita dan informasi yang dapat diterima oleh pelanggan dengan cara *download*.

Karakteristik yang membedakan antara *online* B2C dengan perdagangan konvensional adalah perusahaan dapat berhubungan langsung dengan konsumen tanpa perantara. Hal ini sangat menguntungkan perusahaan dari segi laba karena tidak perlu berbagi keuntungan dengan perantara. Pada awal era *E-Retailing*, perusahaan *retail* seperti *departement store* dan *supermarket* bukan pemain utama dalam *online* B2C, hanya menggunakan *website* untuk memajang produknya, *website* berfungsi sebagai brosur dan menarik minat orang agar datang ketoko mereka. Perusahaan *retail* kemudian mengambil strategi sukses dengan mengombinasi bisnis konvensional mereka dengan bisnis *online* sehingga lahirlah *brick and click store*.

Dewasa ini, banyak pelaku *E-Retailing* bukan berasal dari perusahaan saja, tetapi mulai banyak penjual yang menjual produk melalui toko secara konvensional maupun melalui *internet* secara *online*. Perdagangan seperti ini yang disebut sebagai *traditional retailers with website*, yang berarti suatu bentuk perdagangan *E-Retailing* dimana penjualan dilakukan dengan dua bentuk. Penjual seperti ini kerap juga disebut sebagai *multichannel store.* Ada beberapa faktor yang mempengaruhi keputusan konsumen dalam melakukan pembelian, para *E-tailer* harus mengetahuinya semua kemungkinan untuk menanggapi faktor-faktor tersebut. Ada empat hal yang harus dipenuhi agar konsumen tertarik untuk membeli barang atau menggunakan layanan dari penjual, yaitu :

- Produk, memberikan informasi yang jelas tentang barang yang dijual, lengkap dengan gambar barang tersebut.

- Harga, Menyebutkan harga barang. Di dalam menetapkan harga harus memperhatikan factor lain seperti kualitas, permintaan pasar dan harga pesaing.
- Promosi, mengadakan promosi melalui iklan di media cetak atau elektronik dan alat komunikasi lainnya.
- Pengepakan dan pengiriman, memperhatikan distribusi barang secara fisik dan menjaga keutuhannya. Selain itu pengiriman diusahakan secepat mungkin agar tidak mengecewakan konsumen.

Sukses bisnis *retail* dan *E-Retailing* pada dasarnya adalah menawarkan barang berkualitas dengan harga bersaing, dipadukan dengan pelayanan yang memuaskan. Namun keunggulan *E-Retailing* adalah dapat memperluas pasar sampai keluar negeri dengan investasi nol, sesuatu yang tidak dapat dicapai oleh *retailer* konvensional.

2.2.7 HTML

HTML merupakan singkatan dari *hypertext markup language*, yang merupakan program penulisan informasi pada sebuah *homepage*. Penulisan HTML dapat dilakukan menggunakan alat bantu seperti *Notepad* yang terdapat pada *windows* atau *simple text machintosh*. Selain itu juga dapat digunakan *editor* HTML seperti *Macromedia Dreamweaver* dan *Microsoft Frontpage* yang dapat memudahkan dalam menulis HTML dan memungkinkan dokumen HTML yang dibuat dapat diakses oleh berbagai jenis *Browser*. HTML berupa kode-kode tag yang memberikan instruksi pada *Web Browser* untuk memberikan tampilan sesuai yang diinginkan.

2.2.8 *Cascading Style Sheet (CSS)*

CSS (*Cascading Style Sheets*) merupakan bahasa yang digunakan untuk mengatur tampilan suatu dokumen yang ditulis dalam bahasa *markup/ markup language*. Jika kita berbicara dalam konteks web, bisa diartikan secara bebas sebagai bahasa yang digunakan untuk mengatur tampilan / desain suatu halaman HTML**.** Kelebihan penggunaan CSS adalah halaman web menjadi tidak padat, hingga halaman web menjadi SEO friendly dan kita dapat dengan mudah mengganti tampilan web dengan cukup mengganti nilai dan definisi pada satu file CSS saja.

2.2.9 *JavaScript*

JavaScript adalah bahasa pemrograman berbasis *prototipe* yang berjalan disisi klien. Jika kita berbicara dalam konteks *web*, sederhananya, kita dapat memahami *JavaScript* sebagai bahasa pemrograman yang berjalan khusus untuk di *browser* atau halaman *web* agar halaman *web* menjadi lebih hidup. Kalau dilihat dari suku katanya terdiri dari dua suku kata, yaitu *Java* dan *Script*. *Java* adalah bahasa pemrograman berorientasi objek, sedangkan *Script* adalah serangkaian instruksi program. Dalam pengelolaan pemrograman *JavaScript* yang harus diperhatikan diantaranya *JavaScript* adalah "*case sensitive*", yang artinya *JavaScript* membedakan huruf besar dan huruf kecil.

Kelebihan *JavaScript* adalah dapat bekerja pada sisi *browser*. Maksudnya untuk menampilkan halaman *web*, *user* menuliskan alamat *web* di *address bar url*. setelah itu, *browser* "mengambil" *file html* (dengan *file JavaScript* yang melekat

padanya jika memang ada) ke server yang beralamat di *URL* yang diketikan oleh *user*. Selesai *file* diambil, *file* ditampilkan pada *browser*. Setelah *file JavaScript* berada pada *browser*, barulah *script JavaScript* tersebut bekerja.

Efek dari *JavaScript* yang bekerja pada sisi browser ini adalah *JavaScript* dapat merespon perintah *user* dengan cepat, dan membuat halaman *web* menjadi lebih responsif. *JavaScript* melakukan apa yang tidak bisa dilakukan oleh HTML, PHP, dan CSS yaitu menangani hal – hal yang membutuhkan respon cepat terhadap aksi dari *user*. Contoh : fungsi validasi pada *form*. Ketika mengisikan sebuah *form* yang divalidasi menggunakan *JavaScript*, anda mengetikkan data lalu mengetik *submit*, sebelum data dikirimkan ke *server*, data akan dicek terlebih dahulu pada *browser* menggunakan fungsi *JavaScript* yang ada pada halaman *web*. Sehingga, jika memang data yang anda isikan tidak *valid*, daripada membuang – buang waktu dengan mengirimkan data ke *server* baru di validasi di *server* dan lalu *server* mengirimkan respon balik mengenai ketidak validan *input* data anda, lebih baik cek validasi data *form* dilakukan secara lokal di *browser* menggunakan fungsi *JavaScript*.

2.2.10 *PHP Hypertext Preprocessor*

PHP merupakan sebuah *script open source* multifungsi yang sangat sesuai untuk pengembangan sebuah *website* dan bisa juga digabung kedalam HTML. Hal yang membedakan antara PHP dengan program lain yang sejenisnya adalah bahwa eksekusi dari *code*-nya dilakukan pada sisi *server*. Ini berarti client tidak akan bisa

mengetahui pemrograman seperti apa yang kita buat, berbeda dengan kode dari *Java* yang dieksekusikan pada sisi *client*-nya.

Hal lain yang menjadi salah satu keunggulan lain adalah bahwa PHP sangat mudah untuk dipelajari dan dipahami oleh pemula, sedangkan *programmer* professional akan menjumpai banyak sekali fitur-fitur yang *advance*. Hampir seluruh aplikasi berbasis *web* dapat dibuat dengan PHP, namun fungsi PHP yang paling utama adalah untuk menghubungkan *database* dengan *web*. Dengan PHP, membuat aplikasi *web* yang terkoneksi ke *database* menjadi sangat mudah.

Beberapa sistem *database* yang didukung PHP adalah *Oracle, Sybase, mSQL, MySQL, Solid, Generic ODBC, dan PostgresSQL*. PHP juga mendukung komunikasi dengan layanan lain melalui protocol IMAP, SNMP, NNTP, dan POP3, dan HTTP.

2.2.11 *Database* (Basis Data)

Beberapa definisi tentang Database :

1. Menurut Gordon C. Everest :

   Database adalah koleksi atau kumpulan data yang mekanis, terbagi/shared, terdefinisi secara formal dan dikontrol terpusat pada organisasi.

2. Menurut C.J. Date :

   Database adalah koleksi “data operasional” yang tersimpan dan dipakai oleh sistem aplikasi dari suatu organisasi.

- Data input adalah data yang masuk dari luar sistem

- Data output adalah data yang dihasilkan sistem

- Data operasional adalah data yang tersimpan pada sistem

3. Menurut Toni Fabbri :

   Database adalah sebuah sistem file-file yang terintegrasi yang mempunyai minimal primary key untuk pengulangan data.

4. Menurut S. Attre :

   Database adalah koleksi data-data yang saling berhubungan mengenai suatu organisasi / enterprise dengan macam-macam pemakaiannya.

Jadi SISTEM *DATABASE* adalah sistem penyimpanan data memakai komputer.

2.2.10.1 Sifat-sifat *database*:

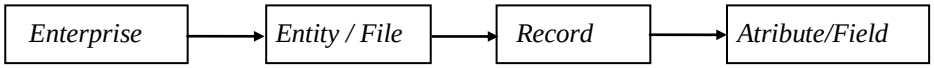
- Internal : Kesatuan (integritas) dari file-file yang terlibat.
- Terbagi/*share*: Elemen-elemen *database* dapat dibagikan pada para *user* baik secara sendiri-sendiri maupun secara serentak dan pada waktu yang sama (*Concurrent sharing*).

2.2.10.2 Elemen-elemen *database*:

A. Tipe:

1. *Enterprise* = Suatu jenis organisasi, misalnya Perusahaan, Bank, Hotel, dan lain-lain.
2. *Entity* = *File* = Obyek pada *enterprise* berdasarkan data yang disimpan.
3. *Atribute* = *Field* = *Data item* = Beberapa hal yang ingin diketahui dari suatu *file*.
4. *Record* = Satu set *field* yang merupakan ciri khas dari suatu *file.*

Istilah *entity* dan *atribute* biasanya digunakan pada tingkat konsepsual dan logikal, sedangkan *file, record* dan *field* pada tingkat *internal*/fisikal. *Enterprise* terdiri dari beberapa *entity, entity* terdiri dari beberapa *record* dan *record* terdiri dari beberapa *field*.



Gambar 2.2 Elemen-elemen Database

B. Isi / Nilai:

1. *Data File* : Seluruh isi data pada *file*
2. *Data Record* : Satu set isi data pada suatu susunan *field* dari suatu *file*
3. *Data Value* : Isi data masing-masing data elemen.

2.2.12 DBMS (*Database Management System*)

Beberapa pengertian tentang DBMS :

1. Menutut C.J. Date : DBMS adalah merupakan *software* yang menghandel seluruh akses pada *database* untuk melayani kebutuhan *user*.
2. Menurut S, Attre : DBMS adalah *software*, *hardware*, *firmware* dan *procedure-procedure* yang *memanage database*. *Firmware* adalah *software* yang telah menjadi modul yang tertanam pada *hardware* (ROM).
3. Menurut Gordon C. Everest : DBMS adalah manajemen yang efektif untuk mengorganisasi sumber daya data.

Jadi DBMS adalah semua peralatan computer (*Hardware+Software+Firmware*). DBMS merupakan *software* (dan *hardware*) yang khusus didesain untuk melindungi dan *memanage database*. DBMS dilengkapi dengan bahasa yang berorientasi pada data (*High level data langauage*) yang sering disebut juga sebagai bahasa generasi ke 4 (*fourth generation language*).

Fungsi DBMS :
- Definisi data dan hubungannya
- Memanipulasi data
- Keamanan dan integritas data
- *Security* dan integritas data
- *Recovery*/perbaikan dan *concurency* data

- Data *dictionary*

- Unjuk kerja / *performance*

Dengan menggunakan DBMS, maka dapat :

- Mendefinisikan data dan hubungannya.
- Mendokumentasikan struktur dan definisi data
- Menggambarkan, mengorganisasikan dan menyimpan data untuk akses yang selektif/dipilih dan efisien.
- Hubungan yang sesuai antara user dengan sumber daya data.
- Perlindungan terhadap sumber daya data akan terjamin, dapat diandalkan, konsisten dan benar.
- Memisahkan masalah Logical dan physical sehingga merubah implementasi database secara fisik tidak menghendaki user untuk merubah maksud data (Logical).
- Menentukan pembagian data kepada para user untuk mengakses secara concurent pada sumber daya data.

Contoh DBMS :

1. Database Hierarchy : Pengaksesan data harus mengikuti aturan hierarchy yang sudah didefinisikan terlebih dahulu.

   Contoh : IMS-2 (Information Management System) oleh IBM, 1968

2. Data Network : Data membentuk jaringan yang lebih bebas dari model hierarchy. Contoh : IDMS (Integrated Database Management System) oleh Cullinett Software Inc, 1972

3. Data Relational : Data dikelompokkan secara bebas menurut jenisnya lewat proses normalisasi.

   Contoh : - INGRES oleh UN of CA & Relational Tech., 1973

   - System-R oleh IBM Research, 1975

   - ORACLE oleh Relational Software Inc. , 1979

   - DBASE II oleh Ashton-Tate, 1981

2.2.13 Kunci Primer, Kunci Tamu dan Index

Key Field adalah sebuah kolom khusus yang memiliki fungsi sebagai pembeda baris satu dengan baris lainnya. *Key* dapat dibagi menjadi dua kategori, yaitu *Primary Key* (kunci primer) dan *Foreign Key* (kunci tamu). Kunci primer adalah suatu kolom (*field*) yang menjadi titik acuan pada sebuah table, bersifat unik, dalam artian tidak ada satu nilai pun yang sama atau kembar dalam table tersebut. Dan dalam satu tabel hanya boleh ada satu kunci primer, jadi benang merah untuk kunci primer adalah **satu** dan **unik**. Sebaliknya, *foreign key* (kunci tamu) adalah sebuah kolom dalam table yang digunakan sebagai "kaitan" bagi tabel lainnya sehingga dibuat hubungan

antara tabel tersebut dengan tabel lainnya. Berbeda dengan kunci primer, kunci tamu bisa terdapat lebih dari satu dalam sebuah tabel dan bisa bersifat tidak unik. *Index* berisi sekumpulan *entri*. Setiap data *record* dan satu *pointer* yang memungkinkan mengakses dengan cepat *record-record* yang diinginkan. Seluruh indeks dikumpulkan lebih kecil dari filenya sendiri, dan indeks selalu dalam keadaan terurut berdasarkan kunci atributnya dimana indeks ini bisa ditelusuri dengan cepat. Pengindeksan akan sangat efektif ketika file-file sangat besar, dimana indeks memerlukan banyak blok. Proses penelusuran dalam indeks yang besar itu sendiri dibantu dengan kembali membuat sunset indeks.

2.2.14 Memahami Normalisasi

Istilah Normalisasi berasal dari E. F.Codd, salah seorang perintis teknologi basis data. selain dipakai sebagai metodologi tersendiri untuk menciptakan struktur tabel 9relasi) dalam basis data (dengan tujuan utnuk mengurangi kemubaziran data) , normalisasi terkadang hanya diipakai sebagai perangkat verifikasi terhadap tabel-tabel yang dihasilkan oleh metodologi lain (misalnya E-R). Normalisasi memberikan panduan yang sangat membantu bagi pengembang untuk mencegah penciptaan struktur tabel yang kurang fleksibel atau mengurangi keflekxibelan.

Kroenke mendefinisikan normalisasi sbagai proses untuk mengubah suatu relasi yang memiliki masalah tertentu ke dalam dua buah relasi atau lebih yang tida memiliki masalah tersebut. Masalah yang dimaksud olej kroenke ini sering disebut dengan istilah anomali.

2.2.15 Model Data *Entity Relationship* (ER Model)

Model ER adalah suatu model yang digunakan untuk pembahasan suatu analisis atau perancangan basis data. Komponen-komponen yang digunakan adalah:

1. Entitas (*entity*)

Adalah individu yang mewakili suatu yang nyata dan dapat dibedakan dari yang lainnya. Sekelompok entitas yang sejenis dan berbeda dalam ruang lingkup yang sama disebut humpunan entitas / entitas set.

2. Atribut (*attributes/ properties*)

Setiap entitas pasti memiliki atribut yang mendiskripsikan karakteristik (*properti*) dari entitas tersebut.

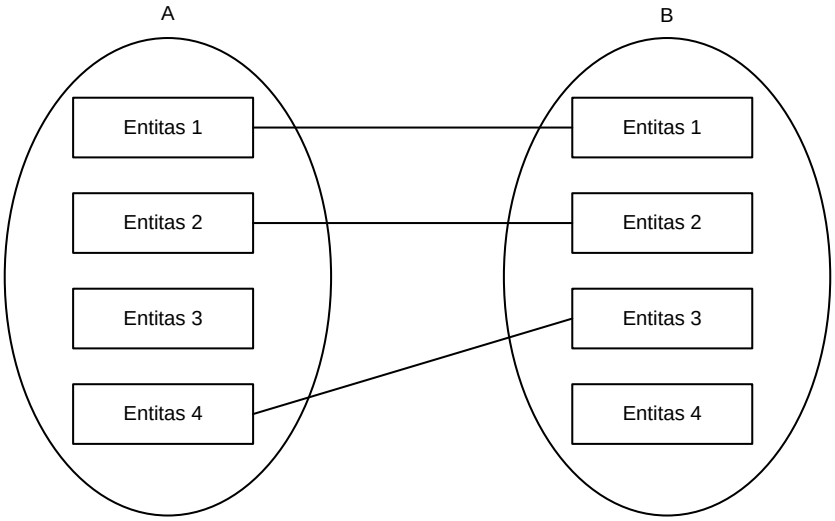
3. Relasi (*Relationship*)

Relasi adalah hubungan antara sejumlah entitas yang berasal dari himpunan entitas yang berbeda.

4. Kardinalitas / derajat

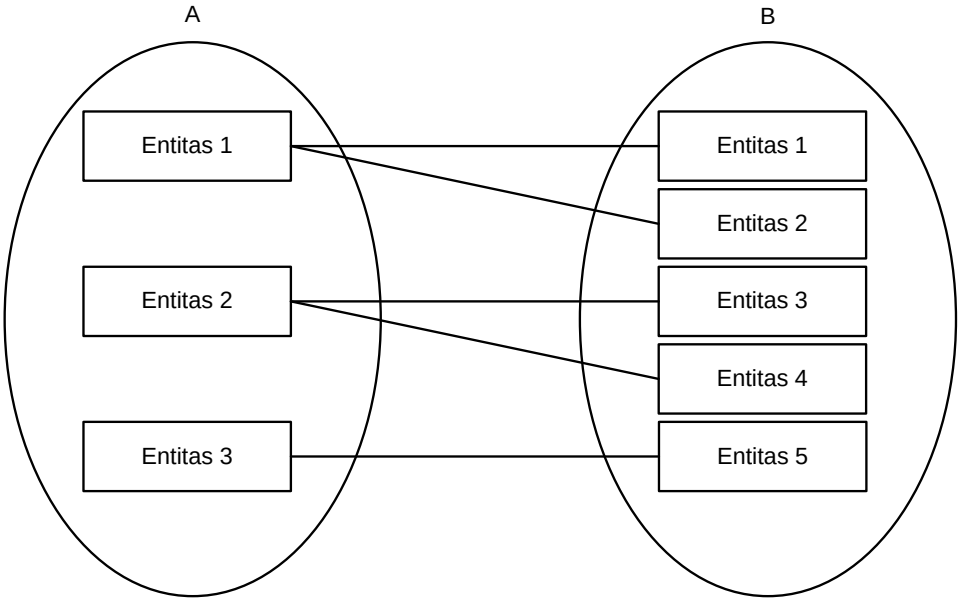
Kardinalitas digunakan untuk menunjukkan jumlah maksimum entitas yang dapat berelasi dengan entitas pada himpunan entitas yang lain. Kardinalitas relasi yang terjadi di antara dua himpunan entitas dapat berupa:

a. satu ke satu (*one to one*), seperti gambar 2.3 berikut ini:



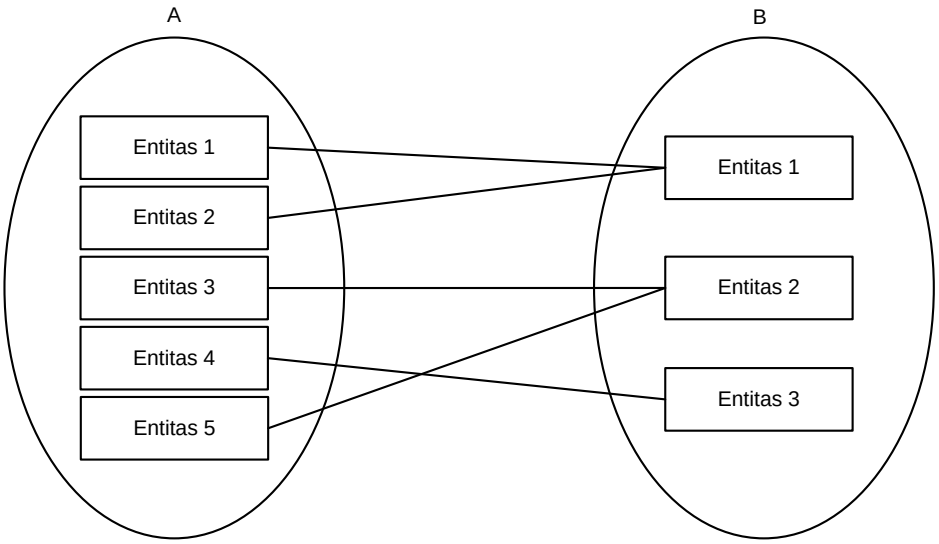
Gambar 2.3 Kardinalitas Relasi Satu ke Satu

b. satu ke banyak (*one to many*), seperti gambar 2.4 berikut ini:



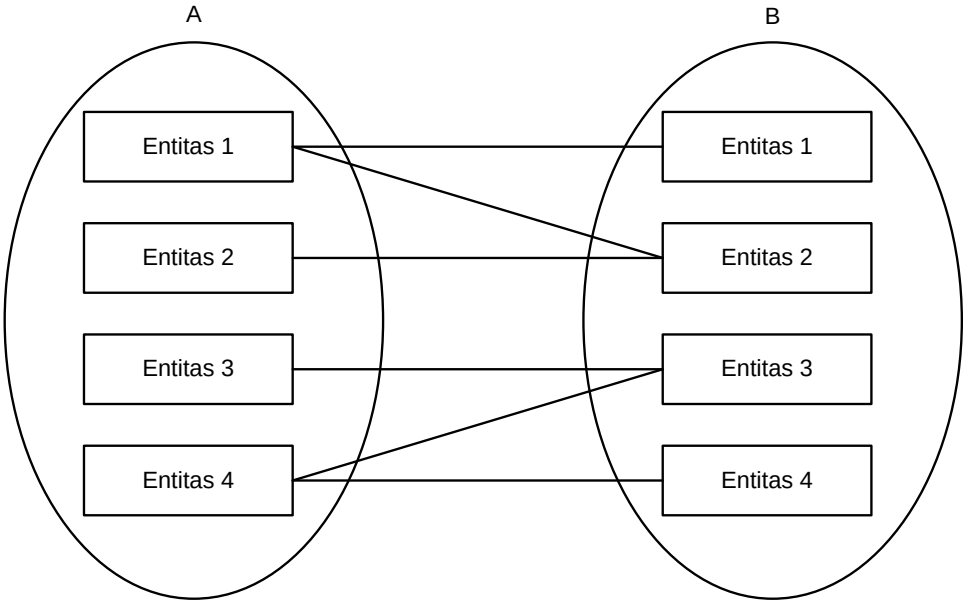
Gambar 2.4 Kardinalitas Relasi Satu ke Banyak

c. banyak ke satu (*many to one*), seperti gambar 2.5 berikut ini:



Gambar 2.5 Kardinalitas Relasi Banyak ke Satu

d. banyak ke banyak (*many to many*), seperti gambar 2.6 berikut ini:



Gambar 2.6 Kardinalitas Relasi Banyak ke Banyak

2.2.16 Diagram ER

Diagram ER digunakan untuk membantu proses perancangan basisdata. Notasi yang digunakan berupa:

1. Persegi panjang (entitas / entity)

   Individu yang mewakili suatu yang nyata dan dapat dibedakan dari yang lainnya.

2. Belah ketupat yang menyatakan himpunan relasi

   Hubungan antara sejumlah entitas yang berasal dari himpunan entitas yang berbeda.

3. Kardinalitas yang dapat dinyatakan dengan garis cabang atau angka
4. Garis sebagai penghubung antara himpunan entitas dengan himpunan entitas yang lain.

2.2.17 *Data Flow Diagram* (DFD)

Data flow diagram (DFD) adalah "Suatu bentuk model yang memungkinkan sistem analisa menggambarkan suatu sistem sebagai suatu jaringan proses dan fungsi yang dihubungkan satu dengan yang lain oleh suatu penghubung. *Data Flow Diagram* (DFD) digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa memperhatikan lingkungan fisik dimana data tersebut mengalir atau disimpan.

Simbol-simbol DFD, antara lain:

a. Simbol Proses

   Menggambarkan bagaimana suatu input ditransformasikan menjadi output. Proses menggambarkan apa yang dilakukan sistem.

b. Simbol *External Entity*

Sering dikenal dengan sumber, tujuan, terminator.*External entity* adalah bagian luar sistem yang dapat men-*supply* input kedalam system dan menggunakan output dari sistem.*External entity* dapat berupa orang/sekelompok orang atau bagian/departemen/pemakai sistem.

c. *Data Store* atau Penyimpanan data

Menggambarkan media penyimpanan data dalam sistem yang berisi data yang digunakan sistem, digunakan sebagai sarana mengumpulkan dan membaca data.*Data Store* dapat merupakan file/database, arsip/catatan manual, lemari/kotak, agenda/buku.

d. Simbol *Data Flow* atau Aliran data

Menggambarkan gerakan paket data/informasi.

2.2.18 Spesifikasi Proses (*Process Spesification* (PSPEC))

Spesifikasi proses (PSPEC) digunakan untuk menggambarkan semua proses modelaliran yang nampak pada tingkat akhir penyaringan. Kandungan dari spesifikasi proses dapat termasuk teks naratif, gambaran bahasa desain program (*Programme Design Language* (PDL)) dari algoritma proses, persamaan matematika, tabel, diagram, atau bagan.

2.2.19 Kamus Data (*Data Dictionary*)

Kamus data (*data dictiionary*) atau disebut juga dengan *systems data dictionary* adalah katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem. Dengan menggunakan kamus dat, analisis sistem dapat mendefinisikan

data yang mengalir di sistem dengan lengkap. Kamus data harus memuat hal-hal berikut ini:

a. Nama arus data
b. Alias atau nama lain dari data dapat dituliskan bila nama lain ini ada. Alias perlu ditulis karena data yang sama mempunyai nama yang berbeda untuk orang atau departemen satu dengan yang lainnya.
c. Bentuk data , dapat berupa dokumen dasar atau formulir, dokumen hasil cetakan komputer, laporan tercetak, tampilan layar di monitor, variabel, parameter, dan *field.*
d. Arus data, menunjukan dari mana data mengalir dan ke mana data akan menuju.
e. Penjelasa, dapat diisi dengan keterangan-keterangan tentang arus data tersebut.
f. Periode, menunjukan kapan terjadinya arus data.
g. Volume, digunakan untuk mengidentifikasikan besarnya simpanan luar yang akan digunakan, kapasitas dan jumlah dari alat input, alat pemroses, dan alat output.
h. Struktur data, menunjukkan arus data yang dicatat di kamus data terdiri dari *item-item* data apa saja.

2.2.20 Skema Relasi

Skema relasi adalah untuk presentasi atribut-atribut dari *entity* yang terdapat dalam sistem dan hubungan antar *entity* pada model ERD. Skema relasi merupakan turunan dari ERD.