BAB II

TINJAUAN PUSTAKA

II.1 Tinjauan Instansi

Tinjauan Instansi di Toko Friska Accessories Cikampek terbagi menjadi empat bagian yaitu riwayat toko, logo toko, struktur organisasi dan deskripsi kerja, visi dan misi perusahaan.

II.1.1 Riwayat Toko

Frisca Acessories Cikampek yang beralamat di Jalan Sukaseuri No. 25 Cikampek Jawa Barat merupakan toko yang bergerak di bidang penjualan produk yaitu *acessories*. Frisca Acessories didirikan pada tanggal 29 Juni 2004 di Cikampek. Pemilik Toko Frisca Acessories adalah Ibu Entin Sartinah, Frisca Acessories mempunyai 5 (lima) karyawan yang terbagi dalam beberapa bagian, seperti bagian administrasi dan keuangan, bagian gudang, dan kasir.

Toko Frisca Accessories sendiri bergerak di bidang penjualan *accessories* dengan mengandalkan merk *accessories* yang sudah cukup terkenal, selain itu Toko Frisca Accessories juga selalu berusaha memberikan pelayanan terbaik kepada konsumen dan selalu memberikan harga yang kompetitif.

II.1.2 Logo Toko

Logo merupakan suatu lambang atau *icon* yang mewakili dari suatu perusahaan atau organisasi, sudah menjadi suatu keharusan bagi perusahaan atau

organisasi untuk mempunyai logo sebagai satu lambang agar perusahaan atau organisasinya lebih dikenal luas. Toko Frisca Accessories memiliki logo seperti terlihat pada gambar II.1 berikut ini:



Gambar II.1 Logo Toko Frisca Accessories Cikampek

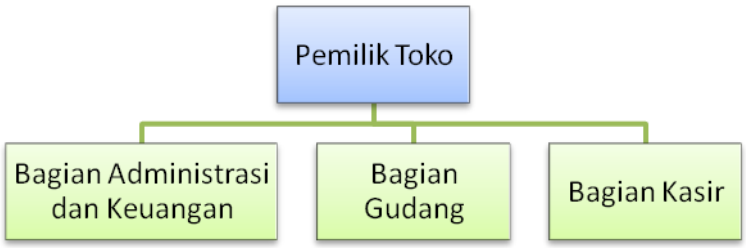
Toko Frisca Accessories Cikampek memiliki logo dengan warna merah dan biru yang berarti memberikan semangat dalam mengembangkan usaha ke arah yang lebih baik serta dapat memberikan kenyamanan kepada karyawan supaya dapat membantu kehidupan karyawan kearah yang lebih baik. Tiga bulatan melambangkan tiga generasi yang berarti Toko Frisca Accessories dapat dipakai oleh anak-anak, remaja, maupun orang tua dan tulisan Frisca Accessories yang berarti nama dari Toko Frisca Accessories Cikampek.

II.1.3 Struktur Organisasi Perusahaan dan Deskripsi Kerja

Struktur organisasi dan deskripsi kerja merupakan hal penting yang harus ada dalam suatu perusahaan karena dengan adanya struktur organisasi dan deskripsi kerja perusahaan lebih terarah dan terencana.

II.1.3.1 Struktur Organisasi

Organisasi adalah sekelompok orang yang bekerja sama untuk mencapai tujuan yang telah direncanakan, dari pengertian tersebut organisasi dapat disimpulkan sebagai kumpulan orang-orang yang bekerja sama dengan dasar persamaan tujuan. Organisasi perusahaan sangatlah penting didalam menjamin kelangsungan dan kelancaran mekanisme kerja perusahaan, adanya organisasi dapat menciptakan suatu sistem pembagian kerja atau tugas yang sesuai dengan kebutuhan perusahaan sehingga mempermudah kegiatan operasional perusahaan dalam mencapai tujuan. Struktur organisasi Toko Frisca Accessories dapat dilihat pada gambar II.2.



Gambar II.2. Struktur organisasi Toko Frisca Accessories

II.1.3.2 Deskripsi Kerja

Deskripsi kerja dari struktur organisasi Toko Frisca Accessories adalah sebagai berikut :

1. Pemilik Toko
    a. bertanggung jawab atas kelangsungan toko,
    b. membuat perencanaan tentang sistem kerja toko,

c. memonitoring perkembangan toko.

2. Bagian Administrasi dan Keuangan mempunyai tugas sebagai berikut :
   a. melaksanakan tugas mencatat segala transaksi yang terjadi,
   b. bertanggungjawab pencatatan laporan, laporan berupa dokumen atau kwitansi,
   c. melakukan pencatatan setiap transaksi yang di toko serta memeriksa kebenaran pencatatan transaksi.
3. Bagian Gudang mempunyai tugas sebagai berikut :
   a. membuat laporan pengeluaran dan pemasukan barang,
   b. bertanggung jawab akan barang-barang yang ada di toko.
4. Bagian Kasir mempunyai tugas sebagai berikut :
   a. bertanggung jawab terhadap pemasukan dan pengeluaran keuangan perusahaan,
   b. membuat anggaran pemasukan dan pengeluaran perusahaan.

II.1.4 Visi dan Misi Perusahaan

Toko Frisca Accessories mempunyai visi dan misi dalam menjalankan tugasnya sebagai salah satu perusahaan yang bergerak dalam penjualan bermacam macam *accessories*, Toko Frisca Accessories sendiri selalu berusaha membalas kepercayaan konsumen dengan *service* yang sangat memuaskan. Adapun visi dan misi Toko Frisca Accessories adalah sebagai berikut:

a. Visi

Visi : "Menjadi toko yang terpercaya dalam menjual produk-produk berupa *accessories*".

b. Misi

Misi : "Meningkatkan nilai aset perusahaan toko melalui peningkatan kualitas SDM, sistem manajemen dan pertumbuhan profit usaha".

II.2 Landasan Teori

Landasan teori merupakan dasar yang digunakan dalam penyusunan *e-commerce* yang akan dibangun. Pada bab ini akan dijelaskan beberapa teori yang berkaitan dengan permasalahan yang akan dibahas sebagai dasar dalam pemahaman dalam sebuah sistem serta metode yang dipakai untuk kegiatan pengembangan terhadap sistem itu sendiri.

II.2.1 Internet

Internet adalah sumber daya informasi yang menjangkau seluruh dunia, *Internet* seringkali disebut *cyberspace* atau *information superhighway*. *Internet* pertama kali dikembangkan dengan nama ARPANET, dikembangkan oleh Departemen Pertahanan Amerika Serikat (*Departement of Defense) tahun 1970* yang bermula untuk keperluan riset di lembaga tersebut. Saat ini jutaan orang telah terhubung ke *Internet* dan saling bertukar informasi melaui aplikasi-aplikasi

seperti *email, web browsing, Ftp (file transfer protocol), chatting* dan lain-lainnya.

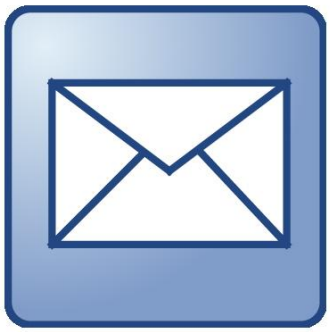
Internet dibangun sebagai sebuah jaringan besar atau global yang menghubungkan banyak (ribuan) jaringan-jaringan kecil, sedangkan yang dimaksud jaringan adalah komputer sebanyak dua atau lebih yang saling berhubungan sehingga dapat bertukar pesan (*exchange*) dan berbagi informasi *(share)*. *Internet* sendiri merupakan sebuah *network* yang paling besar dan telah menjadi solusi dimana memungkinkan setiap individu maupun organisasi yang berada di seluruh penjuru dunia dapat memperoleh informasi serta berkomunikasi tanpa harus mengeluarkan biaya yang besar. Untuk menggunakan layanan *Internet* seseorang dengan perangkat komputer harus terhubung ke *Internet*, baik *dial out* yaitu menggunakan *modem* melalui saluran telepon menghubungi ISP (*Internet Service Provider*), atau melalui jaringan komputer lokal sudah terhubung dengan *Internet*. [2].

II.2.2 Layanan *Internet*

Internet sebagai suatu jaringan komputer mempunyai berbagai kemampuan atau fasilitas bagi para penggunanya. Para pengguna *Internet* dari segala penjuru dunia dapat saling berkomunikasi satu sama lain dengan mempergunakan fasilitas yang ada di *Internet*. Fasilitas tersebut antara lain adalah *email*, FTP (*File Transfer Protocol*), *CHATTING (Real Time Communication), WWW (World Wide Web), HomePage, TCP/IP (Transmission Control Protocol / Internet Protocol), Browser* [2].

II.2.2.1 *Email*

Fasilitas ini merupakan fasilitas yang paling banyak digunakan. Setiap pemakai *Internet* dapat mengirim dan menerima pesan dari orang lain yang juga terhubung dengan *Internet*. Pesan yang dikirim oleh orang tersebut akan diterima oleh penerimanya hampir secara seketika. Keunggulan dari fasilitas ini adalah biaya yang harus dikeluarkan dalam mengirim maupun menerima suatu *email* hampir tidak ada [2]. Lambang *email* dapat terlihat pada gambar II.3.



Gambar II.3 *E-mail*

II.2.2.2 FTP (*File Transfer Protocol*)

FTP adalah suatu protokol dasar yang menyediakan kemampuan untuk menransmisikan *file* dari satu komputer ke komputer yang lain atau dari satu *server* ke *client*.

Berikut ini contoh satu *session* pengambilan sebuah *file* (misalnya laporan.doc) pada sebuah *server* (misalnya ftp.xxx.com) pada direktori /pub/data:

Ftp>Open	⇒	Melakukan Koneksi Ke *Server*
Ftp.Barata.Com		Ftp.xxx.Com

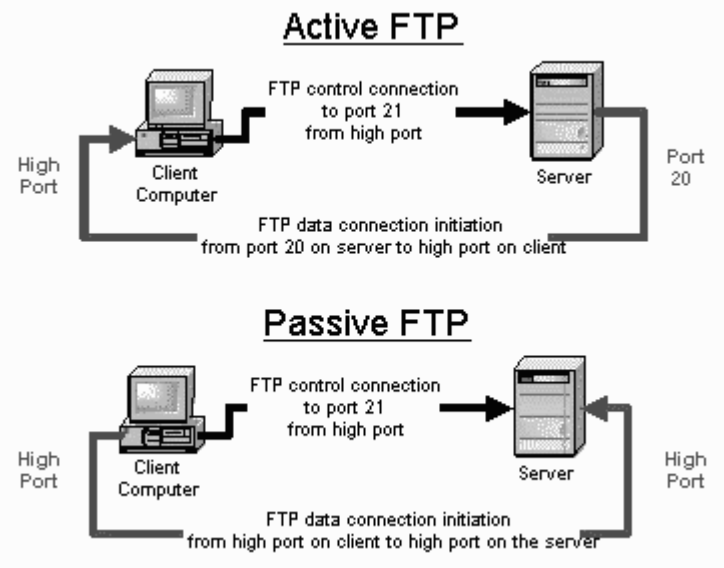
ftp>cd /pub/data ⇒ *Change directory* (pindah ke direktori) /pub/data

ftp>bin ⇒ Ubah mode transfer menjadi *binary*

ftp>get laporan.doc ⇒ Ambil *file* dengan nama rumusan.doc

ftp>Transfer complete ⇒ Status akhir pentransferan *file*

Dari contoh di atas, terlihat bahwa pengguna *Internet* harus mengetahui perintah-perintah yang harus diberikan ke program *ftp* untuk mendapatkan *file* yang diinginkan. Ini akan menjadi *report* jika *file* yang dicari berada pada struktur yang kompleks [2]. Contoh *Ftp* dapat terlihat pada gambar II.4.

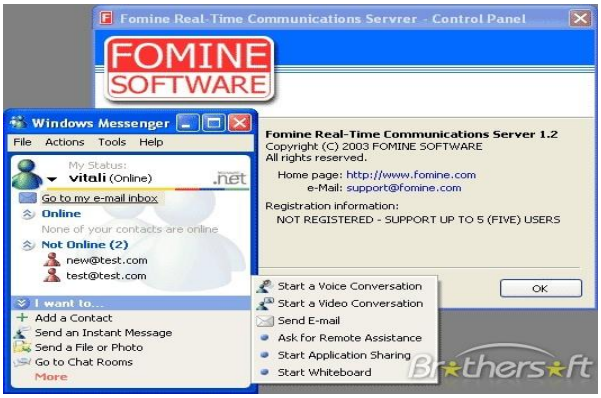


Gambar II.4 *Ftp*

II.2.2.3 *Chatting (Real Time Communication)*

Fasilitas ini berbeda dengan *email* dalam hal pesan yang dikirim oleh pengirim dapat dilihat oleh penerima secara seketika. Penerima kemudian, dapat

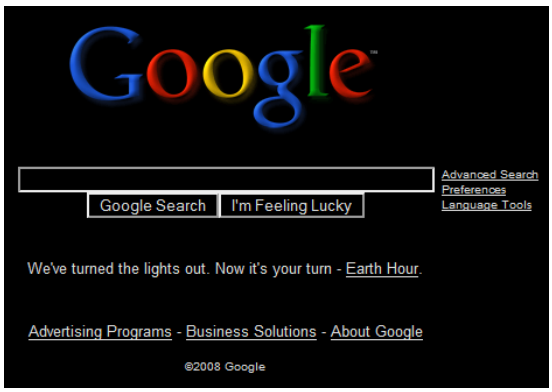
dengan segera membalasnya *(reply)* dan pengirim dapat pula dengan segera membalas *reply* ini [2]. Contoh *Chatting* dapat terlihat pada gambar II.5.



Gambar II.5 *Chatting (Real time Communication)*

II.2.2.4 *HomePage*

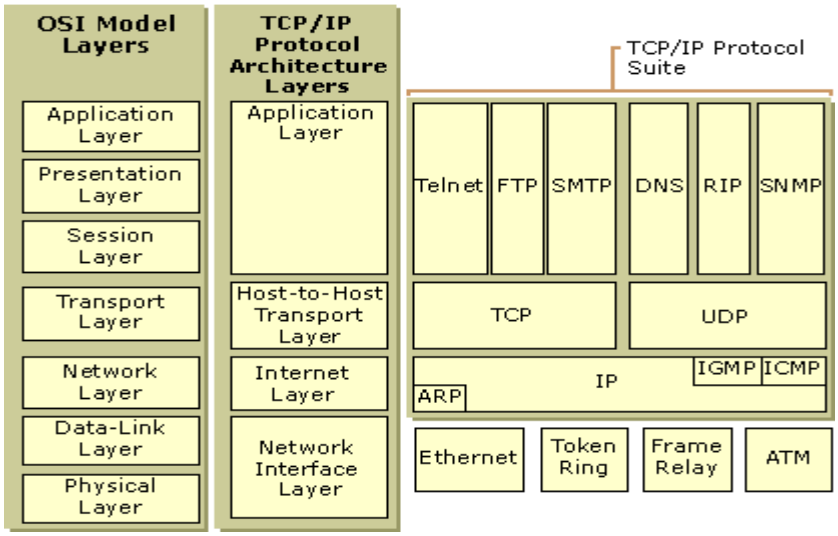
Homepage merupakan halaman yang pertama kali diperoleh saat *netter* memasukkan alamat *web* pada *web browser*. Pada *homepage* terdapat *link-link* yang akan menghubungkan *browser* pada halaman-halaman lain pada *server* yang sama maupun pada *server* di tempat lain. Jadi *homepage* ini boleh dikatakan mirip seperti *menu* dari sebuah *web server* [2]. Contoh *Homepage* dapat terlihat pada gambar II.6.



Gambar II.6 *Homepage*

II.2.2.5 TCP/IP (*Transmission Control Protocol / Internet Protocol*)

TCP/IP (Transmission Control Protocol / Internet Protocol) adalah suatu standar komunikasi data yang digunakan oleh komunitas *Internet* dalam tukar menukar antar *PC*. Standar diperlukan agar antara komputer terjadi kesepakatan tentang tata cara pengiriman dan penerimaan data [2]. Contoh *Ftp* dapat terlihat pada gambar II.7.



Gambar II.7 *TCP/IP (Transmission Control Protocol / Internet Protocol)*

II.2.2.6 *Browser*

Browser adalah sebuah program yang berfungsi untuk menjelajahi halaman-halaman *web* yang terdapat dalam *Internet*. Menjelajahi atau sering disebut sebagai *browsing* adalah suatu aktifitas membuka atau menuju ke situs-situs *(server-server) web* dan membaca informasi yang terdapat didalamnya. Terdapat banyak *browser* yang dapat digunakan tetapi *browser* yang paling banyak digunakan saat ini adalah *Microsoft Internet Explorer (IE), Mozilla Firefox, Opera, Safari, Flock Browser, Google Chrome* dan lain-lain [2]. Contoh *Browser* dapat terlihat pada gambar II.8.



Gambar II.8 *Browser*

II.2.2.7 WWW (*World Wide Web*)

WWW merupakan fasilias di *Internet* yang dikembangkan pada awal 1990 di CERN (*European Laboratory For Particle Physics*). *Web* adalah fasilitas di *Internet* yang sangat memudahkan para pengguna *Internet* berpindah (*jump*) dari satu halaman *web* ke halaman *web* yang lain. *Web* menggunakan suatu protokol

tertentu yaitu HTTP (H*ypertext Transfer Protocol*) untuk mentranmisikan dokumen-dokumen yang dibuat dalam bentuk *Hypertext Markup Language* (*HTML*) dari *server ke client*.

Netter hanya perlu mengetahui satu alamat *server web* yang akan membawanya ke *web server* lain yang memiliki hubungan dengan *server* yang diketahuinya. *Server* ini mungkin saja berada pada negara lain [2].

II.2.3 *Database dan Database System*

Database dan *database system* merupakan hal yang sangat penting dalam pembangunan perangkat lunak, pengertian *database* dan *database system* adalah sebagai berikut :

II.2.3.1 Pengertian *Database*

Database (biasa disebut basis data) adalah kumpulan data yang diorganisasikan dalam tempat penyimpanan *sekunder* berdasarkan aturan tertentu.

a. Operasi Dasar *Database*

Operasi-operasi yang dapat dilakukan berkenaan dengan *database* dapat meliputi :

1. pembuatan *database* baru (*create database*),
2. penghapusan *database* (*drop database*),
3. pembuatan *file*/tabel baru ke suatu *database* (*create table*),
4. penambahan/pengisian data baru ke sebuah *file*/tabel (*insert*),

5. pengambilan data dari sebuah *file*/tabel (*retrieve*),
6. pengubahan data dari *file*/tabel (*update*),
7. penghapusan data dari *file*/tabel (*delete*).

b. Objektivitas *Database*

Secara lebih lengkap, pemanfaatan basis data dilakukan untuk memenuhi sejumlah tujuan (*objektif*) seperti berikut:

1. Kecepatan dan kemudahan (*speed*). Memungkinkan untuk dapat menyimpan atau memanipulasi terhadap data atau menampilkan kembali data tersebut dengan cepat dan mudah.
2. Efisiensi ruang penyimpanan (*space*). Redundansi (pengulangan) data pasti akan selalu ada. Banyaknya redundansi ini akan memperbesar ruang penyimpanan. Dengan *database*, *efisiensi / optimalisasi* penggunaan ruang penyimpanan dapat dilakukan dengan diterapkan sejumlah pengkodean atau dengan membuat relasi-relasi.
3. Keakuratan (*accurancy*). Manfaat pengkodean atau pembentukan relasi antar data dengan penerapan aturan/batasan tipe data, *domain*, keunikan data, dan sebagainya sangat berguna untuk menekan ketidakakuratan pemasukan / penyimpanan data.
4. Ketersediaan (*availbility*). Jumlah data akan semakin bertambah dan tentu membutuhkan ruang penyimpanan yang besar. Untuk itu perlunya dapat memilah adanya data utama/master, data transaksi, dan lain-lainnya. Data

yang sudah jarang digunakan dapat diatur untuk dilepaskan dari *database* yang aktif (*offline*).

5. Kelengkapan (*completeness*). Lengkap atau tidaknya data yang dikelola dalam sebuah *database* bersifat relatif. Karena itu dalam *database* selain data harus juga menyimpan struktur *file*nya untuk mengantisipasi adanya kebutuhan kelengkapan data.
6. Keamanan (*security*). Untuk sistem besar dan serius, aspek keamanan harus diterapkan dengan ketat supaya dapat ditentukan siapa-siapa (pemakai) yang boleh menggunakan *database* dan menentukan operasi-operasi apa saja yang boleh dilakukan.
7. Kebersamaan pemakaian (*sharebility*). *Database* yang dikelola oleh sistem yang mendukung lingkungan *multiuser*, akan dapat memenuhi kebutuhan ini dengan menjaga/menghindari munculnya persoalan baru seperti *inkonsistensi* data (data yang diubah oleh pemakai pada saat yang sama) atau *kondisi deadlock* (banyak pemakai yang saling menunggu untuk menggunakan data).

II.2.3.2 Pengertian Sistem *Database*

Secara umum dapat disimpulkan bahwa sistem *database* adalah sebuah sistem yang terdiri atas kumpulan *file* (tabel) yang saling berelasi atau berhubungan dan sekumpulan program DBMS (*Database Management System*) yang memungkinkan beberapa pemakai atau program lain untuk mengakses dan memanipulasi *file-file* tersebut. *Database* hanyalah sebuah objek yang pasif karena keberadaannya tidak akan berguna jika tidak ada pengelola atau

penggeraknya, yang menjadi penggeraknya secara langsung adalah program (*software*). Gabungan keduanya (*database* dan pengelolanya) menghasilkan sebuah sistem.

Operasi pada suatu sistem *database* dapat diatasi atau dikontrol sesuai dengan kebutuhan. Dalam suatu sistem *database* yang terpadu mutlak diperlukan empat komponen dasar, yaitu:

1. Data

   Data yaitu masukan yang dibutuhkan oleh sistem basis data untuk diolah dan dimodifikasi menjadi informasi yang berguna.

2. *Hardware*

   Hardware adalah peralatan *fisik database* untuk menyimpan dan mengolah data menjadi informasi.

3. *Software*

   Diantara *database physical* dan *end user* terdapat *software* DBMS (*Database Management System*) yang mengatur permintaan akses ke dalam sistem. DBMS adalah kumpulan *file* yang saling berkaitan bersama program pengelolanya.

4. *User*

   User adalah pemakai yang berinteraksi dengan sistem *database* melalui pemanggilan satu program aplikasi permanen.

DBMS (*Database manajemen system*) merupakan perantara bagi pemakai dengan *database* dalam *disk*. Cara berinteraksi antara pemakai dengan *database* tersebut diatur dalam suatu bahasa khusus yang ditetapkan oleh perusahaan

pembuat DBMS. Bahasa tersebut dapat disebut bahasa *database* yang terdiri dari sejumlah perintah (*statement*) yang diformulasikan dan dapat diberikan *user* dan dikenali/diproses oleh DBMS untuk melakukan suatu aksi/pekerjaan tertentu.

Bahasa *database* dapat dipilih ke dalam 2 (dua) bentuk yaitu :

1. *Data Definition Language* (DDL)

   Struktur/skema *database* yang menggambarkan desain *database* secara keseluruhan. Dengan bahasa inilah dapat dibuat tabel baru, membuat indeks, mengubah tabel, menentukan struktur penyimpanan tabel, dan sebagainya.

2. *Data Manipulation Language* (DML)

   Data Manipulation Language (DML) merupakan bentuk bahasa *database* yang berguna untuk melakukan manipulasi dan pengambilan data pada suatu *database*. Manipulasi data dapat berupa penyisipan/penambahan data baru ke suatu basis data, penghapusan data, dan pengubahan data di suatu *database* [2].

II.2.4 SQL (*Structure Query Language*)

DBMS (*Database Management System*) umumnya menyediakan program khusus yang dapat digunakan secara interaktif untuk melakukan berbagai operasi terhadap *database*, seperti pembuatan tabel, penghapusan tabel, penambahan data, mengubah data dan lain-lainnya. Karena DBMS yang digunakan untuk mengelola *database* dan *development tool* yang digunakan untuk menulis aplikasi *database* belum tentu dibuat dari perusahaan yang sama, maka diperlukan sebuah *database* yang sifatnya standar. Dengan bahasa yang standar operasi terhadap *database*

dapat dilakukan dimanapun, tanpa peduli DBMS yang digunakan. SQL mula-mula dibuat pada tahun 1970 dengan *Sequel*. Standarisasi yang pertama dibuat pada tahun 1986 oleh ANSI *(American National Standards Institute)* dan ISO *(International Standards Organization)*, yang disebut sebagai SQL-86. SQL-86 ini diperbaharui pada tahun 1989 menjadi SQL-89. Standar terakhir yang dibuat dan digunakan hingga saat ini adalah SQL-92 yang dikeluarkan pada tahun 1992.

Kendati SQL sendiri merujuk pada *Query Language*, tidak berarti perintah-perintah yang menjadi standar hanyalah perintah-perintah yang berhubungan dengan *query* data. Selain mencakup DML, SQL juga telah dilengkapi dengan berbagai perintah yang tergolong DDL dan perintah kontrol transaksi [2].

II.2.4.1 Struktur Dasar SQL

Ekpresi SQL dasar sebenarnya hanya terdiri atas tiga klausa, yaitu: ***Select, From*** dan ***Where.***

1. Klausa ***select*** digunakan untuk menetapkan daftar atribut (*field*) yang diinginkan sebagai hasil *query*.
2. Klausa **from** digunakan untuk menetapkan tabel yang akan ditelusuri selama *query* data dilakukan.
3. Klausa ***where***, yang sifatnya opsional, digunakan sebagai predikat (kriteria) yang harus dipenuhi dalam memperoleh hasil *query*.

Sintaks (cara penulisan) dari ekspresi SQL dasar dengan tiga klausa tersebut adalah :

Select A1 [, A2,.....,An]

From T1 [, T2.....T2]

[*Where P*]

Keterangan:

A1, A2.....,An merupakan daftar atribut

T1, T2.....,Tn merupakan daftar tabel

P merupakan predikat *query*

[ ] merupakan tanda opsional (boleh digunakan boleh tidak) [2].

II.2.5 *Database* MYSQL

MYSQL merupakan susunan salah satu konsep utama dalam *database* sejak lama, yaitu SQL (*structure Query Language*), kendala dari suatu sistem *database* (DBMS) dapat diketahui dari cara *optimizer*nya dalam melakukan perintah-perintah SQL yang dibuat *user* ataupun program-program aplikasinya. Sebagai *database server* MYSQL dapat dikatakan lebih unggul dibanding *database server* lainnya dalam *query* data. MYSQL adalah salah satu dari sekian banyak sistem *database* yang merupakan terobosan solusi yang tepat dalam aplikasi *database* [4].

MYSQL adalah multi *user database* yang menggunakan bahasa *Structured Query Language* (SQL). MYSQL mampu menangani data yang cukup besar. Perusahaan yang mengembangkan MYSQL yaitu TEX, mengaku menyimpan data lebih dari 40 *database*, 10.000 tabel dan sekitar 7 juta baris, totalnya kurang lebih 100 *Gigabyte* data. SQL adalah bahasa standar yang digunakan untuk

mengakses *database server*. Bahasa ini pada awalnya dikembangkan oleh IBM, namun telah diadopsi 25 dan digunakan sebagai standar industri.

Dengan menggunakan SQL, proses akses *database* menjadi lebih *user-friendly* dibandingkan dengan menggunakan *dBase* atau *Clipper* yang masih menggunakan perintah-perintah pemrograman. MYSQL merupakan *software database* yang paling populer di lingkungan Linux, kepopuleran ini karena ditunjang performa *query* dari *database*nya yang saat ini bisa dikatakan paling cepat dan jarang bermasalah. MYSQL ini juga sudah dapat berjalan pada lingkungan *Windows.*

Perintah untuk mengelola *database* dibagi menjadi 3 (tiga) kelompok, diantaranya:

1. Perintah untuk mendefinisikan data/DDL (*Data Definition Language*).
2. Perintah untuk memanipulasi data/DML (*Data Manipulation Language*).
3. Perintah untuk mengendalikan data/DCL (*Data Control Language*).

II.2.6 *Hyper-Text MarkUp Language* (HTML)

Dokumen *HTML* adalah *file* teks regular (disebut juga ASCII) yang diciptakan dengan menggunakan editor teks (misalnya, *Emacs* dalam *Unix*, *Gnotepad* pada *Linux*, atau *Notepad* pada *Windows*) atau dengan menggunakan *word processor*, tetapi harus disimpan dengan *format "text only with line breaks".* Sesuai dengan namanya, bahasa ini menggunakan tanda (*MarkUp*) untuk menandai perintah-perintahnya. Saat ini banyak sekali aplikasi-aplikasi yang dapat digunakan untuk membuat *web page*, seperti *Microsoft Frontpage*, *Adobe*

GoLive, *Macromedia Dreamweaver*, *Ultradev* dan lain-lainnya. Namun untuk seorang *web develover* kemampuan dasar perintah *HTML* tetap dibutuhkan [2].

II.2.7 PHP

PHP adalah bahasa *server-side scripting* yang menyatu dengan HTML untuk membuat halaman *web* yang dinamis. Maksud dari *server-side scripting* adalah sintaks dan perintah-perintah yang diberikan akan sepenuhnya dijalankan di *server* tetapi disertakan pada dokumen HTML [4]. PHP merupakan *script* untuk pemrograman *script web service side*, *script* yang membuat dokumen HTML secara *on the Fly*, dokumen HTML yang dihasilkan dari suatu aplikasi bukan dokumen HTML yang dibuat menggunakan editor teks atau editor HTML. Dengan menggunakan PHP maka *maintenance* suatu situs *web* menjadi lebih mudah. Proses *Update* data dapat dilakukan dengan menggunakan aplikasi yang dibuat dengan menggunakan *script* PHP. Kemampuan (*feature*) PHP yang paling diandalkan dan signifikan adalah dukungan kepada banyak *database*. Membuat halaman *web* yang menggunakan data dari *database* dengan sangat mudah dapat dilakukan. Berikut adalah daftar *database* yang didukung oleh PHP Adabas D, dBase, Empress, *File* pro (*read olny*), FrontBase, Hyperwave, IBM DB2, Informix, Interbase, MSQL, MYSQL, Oracle (OC17 dan OC18) dan lain-lain.

PHP merupakan *software* yang *open source* (gratis) dan mampu lintas *platform*, yaitu dapat digunakan dengan sistem operasi dan *web server* apapun. PHP mampu berjalan di *Windows* dan beberapa versi Linux. PHP juga dapat dibangun sebagai modul pada *web server* Apache dan sebagai *binary* yang dapat

berjalan sebagai CGI. PHP dapat mengirim HTTP *Header*, dapat mengeset cookies, mengatur *authentucation* dan *redirect users*. PHP menawarkan koneksitas yang baik dengan beberapa basis data, antara lain Oracle, Sysbase, MSQL, MYSQL, Solid, PostgreSQL, Adabas, Filepro, Velocis, dBase, Unix dbm, dan tak terkecuali semua *database* ber-*interface* ODBC. Juga dapat berintegrasi dengan beberapa *library eksternal* yang dapat melakukan segalanya mulai dari membuat dokumen PDF hingga mem-*parse* XML.

PHP juga mendukung komunikasi dengan layanan lain melalui *Protocol* IMAP, SNMP, NNTP, POP3 atau bahkan HTTP. Bila PHP berada dalam halaman *web*, maka tidak lagi dibutuhkan pengembangan lingkungan khusus stau direktori khusus. Hampir seluruh aplikasi berbasis *web* dapat dibuat dengan PHP. Namun kekuatan utama adalah konektivitas basis data dengan *web*. Dengan kemampuan ini kita mempunyai suatu sistem basis data yang dapat diakses dari *web*.

Pembuatan *web* ini merupakan kombinasi antara PHP sendiri dan sebagai bahasa pemrograman dan HTML sebagai pembangun halaman *web.* Ketika seorang pengguna *Internet* akan membuka situs yang menggunakan fasilitas *server-side scripting* PHP di *server* lalu mengirimkan hasilnya dalam format HTML ke *web browser* pengguna *Internet* tadi. Dengan demikian seorang pengguna *Internet* tidak dapat melihat kode program yang ditulis dalam PHP sehingga keamanan dari halaman *web* menjadi lebih terjamin, tetapi tidak seperti ASP yang cukup dikenal sebagai *server-side scripting.*

II.2.8 *Java Script*

JavaScript dikembangkan oleh Netscape, yang merupakan bahasa pemrograman yang sederhana karena tidak bisa digunakan untuk membuat suatu aplikasi atau pun applet. Java Script merupakan sebuah bahasa *scripting* yang dikembangkan oleh Netscape. Dengan Java Script kita bisa dengan mudah membuat sebuah halaman *web* yang interaktif. Program Java Script dituliskan pada *file* HTML (*.htm atau *.html) dengan menggunakan *tag* kontainer <SCRIPT>. Dengan kata lain, Anda tidak perlu menuliskan program JavaScript pada *file* terpisah (walaupun Anda juga bisa melakukannya) [13]. *JavaScript* merupakan bahasa *scripting* yang popular di *Internet* dan dapat bekerja di sebagian besar *browser* popular seperti *Internet Explorer* (IE), *Mozilla Firefox*, *Netscape dan Opera*. Kode *JavaScript* dapat disisipkan dalam halaman *web* menggunakan tag *SCRIPT*.

II.2.9 CSS (*Cascading Style Sheets*)

CSS adalah singkatan dari *Cascading Style Sheets.* CSS bisa diartikan aliran dari suatu kode ke kode lain yang saling berhubungan. CSS juga merupakan kumpulan kode-kode yang berurutan dan saling berhubungan untuk mengatur format/tampilan suatu halaman *HTML*.

II.2.9.1 Sejarah CSS (*Cascading Style Sheet*)

Pada tanggal 17 Agustus 1996, *World Wide Consortium* (W3C) menjadikan CSS sebagai bahasa pemrograman standart dalam pembuatan dokumen *web*.

Tujuannya adalah mengurangi pembuatan tag-tag baru oleh Netscape dan *Internet Explorer*, karena kedua kedua *browser* pada saat itu bersaing mengembangkan Tag sendiri untuk mengatur tampilan *web*.

CSS level 1 mendukung pengaturan tampilan dalam hal:

1. *Font* (jenis, ketebalan)
2. Warna, *teks*, *background* dan elemen lainnya
3. *Text attributes*, misalnya spasi antar baris, kata dan huruf
4. Posisi *teks*, gambar, *table* dan elemen lainnya
5. *Marjin*, *border* dan *padding*.

Selanjutnya di tahun 1998, W3c menyempurnakan CSS awal dengan menciptakan standar CSS2 (CSS level 2) yang menjadi standar hingga saat ini. Pada CSS level 2 ini, dimasukkan semua atribut dari CSS1, serta diperluas dengan penekanan pada *Internasional accessibility and capability* khususnya *media-specific* CSS. Bahkan pada tahun 2000, tidak lama setelah CSS2 diimplementasikan. CSS2 ini sampai sekarang masih terus dikembangkan, spesifikasinya dibagi pada beberapa topik atau modul.

II.2.9.2 Istilah-Istilah dalam CSS (*Cascading Style Sheets*)

Istilah-istilah dalam CSS ada 3 (tiga), yaitu *style rule*, *style sheet*, dan *selector*, sedangkan *selector* terbagi menjadi 2 (dua) bagian yaitu *selector* dan *declaration.*

1. Style rule

Cascading style sheet merupakan kumpulan aturan yang mendefinisikan *style* dari *document*. Sebagai contoh kita bisa membuat aturan *style* yang menentukan bahwa semua <H2> ditampilkan dengan warna *orange*.

2. Style sheet

Style sheet dapat dapat di*embedded* ke *HTML document*. Atau disebut *embedded style sheet. Style sheet* juga bisa dibuat sebagai external *file* dan di link ke dokument *HTML*. *Style role* bisa di kenakan pada bagian tertentu dari *web page*. Sebagai contoh anda bisa menentukan *paragraph* tertentu ditampilkan dengan *style bold* dan *italic* sementara yang lain tetap seperti biasa.

3. Selector

selector { property1: value; property2:value, . . .} H1{ color:green; background-color:orange} Style sheets terdiri dari dua bagian:

a). *Selector*

Bagian pertama sebelum tanda "{}" disebut *selector*

b). *Declaration*

Terdiri dari *property* dan nilainya.

II.2.9.3 Keuntungan Menggunakan CSS (*Cascading Style Sheets*)

1. *Update* tempilan lebih mudah

Meng*update* tampilan situs *web* yang terdiri dari beberapa halaman saja tentu jauh lebih mudah daripada untuk situs dengan ratusan atau ribuan halaman. Dengan CSS, kita akan dapat mengubah tampilan halaman-halaman *web* dalam

jumlah banyak dengan mudah, karena semua tag untuk *style* ini berada pada suatu *file* CSS saja.

2. Beban *bandwidth* lebih kecil

Dengan CSS, ukuran *file web* akan menjadi kecil karena *tag-tag style* dipisahkan secara mandiri. Hal ini secara signifikan akan berdampak pada proses *loading* yang lebih cepat. Selain itu, CSS hanya akan di-*load* satu kali oleh *browser* dan akan terus diterapkan pada halaman-halaman lain. Hal ini berbeda dengan situs yang menggunakan *table* karena harus di-*load* berulang-ulang.

3. Modifikasi *web template* lebih mudah

Cara paling cepat membangun situs *web* ialah dengan menggunakan *template*. Namun terkadang *template* tersebut tidak sesuai dengan keinginan dan keperluan kita. Dengan CSS, kita akan lebih mudah melakukan perubahan pada *template* tersebut, misalnya mengganti warna latar atau mengubah jenis huruf.

4. Lebih mudah digunakan pada *mobile phone*

Sekarang, kegiatan *browsing* tidak hanya melalui *browser* di komputer atai PC saja, tetapi juga melalui *mobile phone* atau PDA. Situs *web* yang menggunakan CSS akan jauh lebih mudah digunakan oleh mereka yang browsing melalui *gadget-gadget* tersebut.

II.2.9.4 Kerugian menggunakan CSS (*Cascading Style Sheets*)

Tidak semua *browser* mengartikan kode CSS dengan cara yang sama. Jadi kadang-kadang tampilan *web* dengan CSS terlihat baik di *browser* yang satu, tapi berantakan di *browser* yang lain. Jadi harus memeriksa tampilan supaya terlihat baik di semua *browser* dan menambahkan kode-kode khusus *browser* tertentu jika memang dibutuhkan agar tampilan *website* terlihat baik disemua *browser* [2].

II.2.10 Adobe Dreamweaver

Adobe Dreamweaver merupakan program penyunting halaman *web* keluaran *Adobe Systems* yang dulu dikenal sebagai *Macromedia Dreamweaver* keluaran *Macromedia*. Program ini banyak digunakan oleh pengembang *web* karena *fitur-fiturnya* yang menarik dan kemudahan penggunaannya. Versi terakhir *Macromedia Dreamweaver* sebelum *Macromedia* dibeli oleh *Adobe Systems* yaitu versi 8. Versi terakhir *Dreamweaver* keluaran *Adobe Systems* adalah versi 10 yang ada dalam *Adobe Creative Suite 4* (sering disingkat *Adobe CS4*) [2].

II.2.11 *E-commerce*

Perdagangan elektronik atau *e-commerce* adalah penyebaran, pembelian, penjualan, pemasaran barang dan jasa melalui sistem elektronik seperti *Internet* atau televisi, *www*, atau jaringan komputer lainnya. *e-commerce* dapat melibatkan *transfer* dana elektronik, pertukaran data elektronik, sistem manajemen inventori otomatis, dan sistem pengumpulan data otomatis.

Industri teknologi informasi melihat kegiatan *e-commerce* ini sebagai aplikasi dan penerapan dari *e-bisnis* (*e-business*) yang berkaitan dengan transaksi komersial, seperti: *transfer* dana secara elektronik, SCM (*supply chain management*), pemasaran secara elektronik (e-*marketing*), atau pemasaran *online* (*online marketing*), pemrosesan transaksi *online* (*online transaction processing*), pertukaran data elektronik (*electronic data interchange* /EDI) dan lain lain.

E-commerce merupakan bagian dari e-*business*, di mana cakupan e-*business* lebih luas, tidak hanya sekedar perniagaan tetapi mencakup juga pengolaborasian mitra bisnis, pelayanan nasabah, lowongan pekerjaan dan lain-lain. Selain teknologi jaringan *www*, *e-commerce* juga memerlukan teknologi basis data atau pangkalan data (*databases*), surat elektronik (e-*mail*), dan bentuk teknologi *non* komputer yang lain seperti halnya sistem pengiriman barang, dan alat pembayaran untuk *e-commerce* ini.

E-commerce pertama kali diperkenalkan pada tahun 1994 pada saat pertama kali *banner-elektronik* dipakai untuk tujuan promosi dan periklanan di suatu halaman-*web* (*website*). Menurut *Riset Forrester*, perdagangan elektronik menghasilkan penjualan seharga AS$12,2 milyar pada 2003. Menurut laporan yang lain pada bulan Oktober 2006 yang lalu, pendapatan ritel *online* yang bersifat *non-travel* di Amerika Serikat diramalkan akan mencapai seperempat trilyun dolar US pada tahun 2011.

II.2.11.1 Elemen didalam *E-commerce*

Elemen-elemen yang terdapat dalam *e-commerce* adalah sebagai berikut:

1. *Networking* atau jaringan merupakan beberapa komputer yang saling berhubungan dan saling tukar menukar informasi dan terkoneksi melalui sebuah kartu jaringan dan jalur komunikasi, yang terdiri dari jaringan *local* atau lebih dikenal dengan *LAN* dan jaringan *Internet* yang banyak digunakan untuk jaringan *LAN* atau lokal terdapat berbagai jenis jaringan seperti jaringan *bus, token, star* dan lain-lain yang cocok untuk digunakan.
2. *Security* (keamanan), *security* atau keamanan merupakan bagian penting, karena menyangkut masalah keamanan data *member* dan juga keamanan server, masalah seperti *enkripsi* data pada saat transaksi memakai kartu kredit, aplikasi yang sudah banyak dipakai yaitu *SSL (Secure Socet Layer).*
3. *Web programming* dan *web design* Kedua masalah ini dalam pelaksanaannya dapat dikerjakan oleh satu orang atau lebih, untuk *web programming* dikhususkan dalam pembuatan bahasa pemrograman. Untuk *web design*, khusus untuk mendesain halaman *web* atau peraturan gambar, warna maupun tata letak suatu *web* dari segi keindahan dan dinamis. *Web programming* biasanya menggunakan *script server* seperti PHP, ASP, CGI dan yang lainnya. Untuk designnya banyak *software-software* yang mendukung seperti *Macromedia, photosop, Frontpage, Office publisher* dan masih banyak lagi yang lainnya.
4. *Business Online* maksud dari *bussines online* disini yaitu menyangkut bagaimana cara pengolahan perusahaan mulai dari *manajemen, administrasi*

keuangan dan lainnya, yang semua itu merupakan suatu strategi dalam menarik *customer* atau konsumen untuk datang ke Toko Frisca Accessories.

5. *Online Payment* merupakan metode pembayaran secara *online*. Pembayaran *online* yang sering digunakan oleh beberapa sistem *e-commerce* yaitu menggunakan jasa pembayaran *Paypal. Paypal* adalah salah satu alat pembayaran *(Payment procesors)* yang terbanyak digunakan dan teraman. Pengguna *Internet* dapat membeli barang di *e_bay, lisensi software original*, keanggotaan situs, urusan bisnis, mengirim dan menerima donasi/sumbangan, mengirim uang ke pengguna *Paypal* lain di seluruh dunia dan banyak fungsi lainnya dengan mudah dan otomatis menggunakan *Internet* atau *mobile. PayPal* mengatasi kekurangan dalam pengiriman uang tradisional seperti cek atau *money order* yang prosesnya dapat memakan waktu lama.
6. *Cyberlaw* adalah hukum yang digunakan di dunia *cyber* (dunia maya) yang umumnya diasosiasikan dengan *Internet*. *Cyberlaw* merupakan aspek hukum yang ruang lingkupnya meliputi setiap aspek yang berhubungan dengan orang perorangan atau subyek hukum yang menggunakan dan memanfaatkan teknologi *Internet* yang dimulai pada saat mulai *online* dan memasuki dunia *cyber* atau maya. *Cyberlaw* sendiri merupakan istilah yang berasal dari *Cyberspace Law*. *Cyberlaw* akan memainkan peranannya dalam dunia masa depan, karena nyaris tidak ada lagi segi kehidupan yang tidak tersentuh oleh keajaiban teknologi dewasa ini dimana kita perlu sebuah perangkat aturan main didalamnya (*virtual world*) [2].

II.2.11.2 Klasifikasi *E-commerce*

Menurut Taryana Suryana [5], *e-commerce* diklarifikasikan ke dalam beberapa model yaitu:

1. Model *Store Front*

Store front ialah kombinasi proses transaksi, *security*, pembayaran secara *online* serta penyimpanan informasi yang memungkinkan para pedagang untuk menjual dagangannya di *Internet* melalui *website*. *Store front* merupakan konsep dasar perdagangan elektronik dimana terjadi interaksi penjual dan pembeli secara langsung. Untuk menjalankan *store front*, seseorang harus mengorganisasikan dalam bentuk katalog produk yang ditempatkan pada halaman *website*. Pemilik bisnis *online* juga harus dapat menerima pembayaran secara *online* dalam kondisi aman, mengatur pengiriman dagangan para konsumen serta mengolah data konsumen. Dalam praktiknya, para pedagang *online* menggunakan teknologi yang disebut *shopping cart* atau kereta belanja. Dengan menggunakan *shopping cart* konsumen dapat memasukkan barang barang pesanannya kedalam *shopping cart* nya.

2. Model Lelang

Model lain perdagangan *online* ialah model lelang, model ini berfungsi sebagai forum dimana para pengguna *Internet* dapat memasuki *website* dengan berperan sebagai penawar atau penjual. Jika seseorang berperan sebagai penjual produk, maka yang bersangkutan dapat mencari situs-situs yang menyediakan

barang-barang yang sedang dicari, melihat lihat kegiatan penawaran saat itu dan memasang penawaran.

3. Model Portal

Model portal merupakan bentuk lain dari e-*commerce*. Portal berisi berbagai informasi meliputi mulai berita politik dan ekonomi, olahraga, teknologi, sampai dengan berita-berita ringan mengenai kehidupan selebriti, gaya hidup, cerita bersambung dan lain-lain.

4. Model *Dynamic Pricing*

Model *dynamic pricing* atau penetapan harga merupakan model yang mengikuti pola mekanisme bisnis, yaitu bagaimana antara bisnis berlangsung dan produk diberi harga, dengan demikian seorang konsumen dapat membeli suatu produk dengan tawaran yang paling rendah. Strategi lain ialah dengan cara menawarkan produk atau jasa tertentu secara gratis. Model dalam kategori ini ialah model menentukan sendiri harga produk (*your price model*), model harga perbandingan (*comparison pricing model*), model harga *sensitive* didasarkan kebutuhan (*demand sensitive pricing model*), model *barter*, model *rebate* dan model penawaran produk dan jasa secara gratis.

5. Model *Online Trading*

Model *online trading* biasanya merupakan perdagangan elektronik dalam bentuk perdagangan sekuritas. Perdagangan saham secara *online* dilakukan oleh

broker yang mendapatkan komisi karena jasa para broker dalam mengatur jalannya perdagangan saham tersebut. Melalui perdagangan saham secara *online* ini, konsumen dapat melakukan penelitian terhadap sekuritas, membeli dan menjual investasi melalui komputer yang tersambung dengan *Internet*.

6. Model *Online Loan*

Online loan atau pinjaman secara *online* merupakan salah satu bisnis *online* yang sudah cukup popular, konsumen saat ini dapat mencari pinjaman dengan bunga rendah melalui *Internet*. Salah satu portal yang *member* pinjaman secara *online* adalah *e-loan* dengan alamat *www.eloan.com*. Situs ini menawarkan layanan kartu kredit, pinjaman pembelian rumah dan peralatan, serta kalkulator untuk membuat konsumen mahir dalam memutuskan mencari pinjaman.

II.2.12 *Paypal*

Paypal adalah salah satu alat pembayaran (*payment procesors)* menggunakan *Internet* yang terbanyak digunakan di dunia dan teraman. Pengguna *Internet* dapat membeli barang di *e-bay*, lisensi *software* original, keanggotaan situs, urusan bisnis, mengirim dan menerima donasi/sumbangan, mengirim uang ke pengguna *Paypal* lain di seluruh dunia dan banyak fungsi lainnya dengan mudah dan otomatis menggunakan *Internet* atau *mobile*, *Paypal* mengatasi kekurangan dalam pengiriman uang tradisional seperti cek atau *money order* yang prosesnya dapat memakan waktu *PayPal* seperti rekening bank, pertama anda membuat *account*, lalu dana akan ditarik langsung dari kartu kredit anda ke

account tersebut atau dengan dana dari hasil transfer *account PayPal* orang lain ke Saldo / *Balance PayPal* anda, dan anda sudah dapat menggunakan *account* PayPal untuk bertransaksi [2].

II.2.13 *Search Engine Optimization* (SEO)

SEO (*Search Engine Optimization*) adalah serangkaian proses yang dilakukan secara sistematis yang bertujuan untuk meningkatkan volume dan kualitas trafik dari mesin pencari menuju alamat situs *web* tertentu dengan memanfaatkan mekanisme kerja alami algoritma mesin pencari tersebut [6].

Praktisi SEO terkenal seperti *Rand Fishkin, Barry Schwartz, Aaron Wall and Jill Whalen* mencoba mempelajari beberapa pendekatan berbeda untuk *Search Engine Optimization*, dan telah menerbitkan opini mereka di forum *online* dan blog. Praktisi SEO mencoba mempelajari pola yang diterapkan oleh beragam *Search Engine* untuk melihat lebih dalam algoritma tersebut.

Berdasarkan *Jerri L. Ledford* (*Search Engine Oprimization*), SEO adalah teknik pencarian yang menggunakan kata kunci atau *frase* yang mengandung indikator yang terkandung dalam halaman-halaman *web*, informasi tersebut yang akan diindex oleh mesin pencari [7].

Berdasarkan *David Viney* (*Get to the top on Google*), SEO adalah sebuah teknik pengoptimasian sebuah halaman yang mengandung kata kunci atau *frase* yang bersangkutan yang akan diindex oleh mesin pencari sebagai kata kunci pencarian [8].

Berdasarkan *Peter Kent* (*Search Engine Oprimization for Dummies*), SEO adalah teknik yang digunakan untuk mendapatkan posisi yang menguntungkan di mesin pencari yang sesui dengan kata kunci yang berada di halaman *website*.

Tujuan dari SEO adalah agar *web/blog* selalu berada di halaman terdepan dan teratas dari suatu *Search Engine*, sehingga selalu berada di posisi teratas, maka besar kemungkinan *web* atau blog sering dikunjungi.

II.2.13.1 Sejarah SEO

Pada pertengahan tahun 1990-an ketika para webmaster dan penyedia konten mulai mengoptimalkan situs mereka di *Search Engine* yang mulai menggunakan katalog. Cara optimasi situs (SEO) saat itu adalah dengan mengirimkan alamat sebuah situs atau halaman situs dalam bentuk URL ke berbagai *search*. Pada periode tersebut algoritma *Search Engine* belum terlalu kompleks sehingga mudah dimanipulasi. Versi awal algoritma pencarian didasarkan sepenuhnya pada informasi yang diperoleh pada saat "*crawling*" oleh "*spider*" pada bagian "*meta tag*" pada kode html situs *web*. "*Meta tag*" menyediakan informasi tentang konten yang terkandung pada suatu halaman *web* dengan serangkaian *keyword*. Bagian inilah yang kemudian dimanipulasi dengan cara memasukkan *keyword* yang tidak relevan, tidak lengkap, dan tidak konsisten dengan konten situs, sehingga *Search Engine* salah menempatkan dan memeringkat situs tersebut. Hal ini menyebabkan hasil pencarian menjadi tidak akurat dan menimbulkan kerugian baik bagi *Search Engine* maupun bagi pengguna *Internet* yang mengharapkan informasi yang relevan dan berkualitas.

Search Engine yang yang dikembangkan oleh *Larry Page* dan *Sergey Brin*, dua orang mahasiswa doktoral ilmu komputer *Universitas Stanford*, cukup berhasil mengatasi permasalahan tersebut dengan mengembangkan *Backrub*, sebuah *Search Engine* yang sepenuhnya mengandalkan algoritma matematis untuk memeringkat halaman *web*.

Bisnis dan layanan SEO berkembang pesat seiring dengan pertumbuhan *web*, yang menyebabkan suatu situs *web* harus berjuang sekuat tenaga agar urlnya mudah ditemukan calon pelanggan di antara jutaan alamat situs lain dari seluruh dunia yang menjadi kompetitornya. *Search Engine* merupakan pintu masuk utama, karena pengguna internet tidak lagi sanggup (dan tidak ingin) menghafalkan alamat-alamat situs *web*, dan sebagai gantinya mereka mengandalkan *indeks* yang dibuat oleh *Search Engine* [8].

II.2.13.2 Fungsi SEO

Fungsi dari SEO antara lain :

a. Memberikan posisi terbaik di mesin pencari.

b. Memberikan *traffic* sesui dengan target pengunjung

c. Mempopulerkan situs *website* dengan cepat.

II.2.13.3 Kelebihan SEO

Dengan adanya mesin pencari saat ini, maka *user* akan dipermudah dalam mencari dan mengetahui situs – situs yang akurat sesuai dengan informasi yang dibutuhkan oleh *user*. SEO dapat memberikan pengunjung informasi yang lebih akurat yang sesuai dengan isi *content* situs serta mempermudah *user* untuk

mendapatkan informasi yang akurat dan sesuai dengan informasi yang dibutuhkan.

II.2.13.4 Kekurangan SEO

Banyaknya situs *web* di seluruh dunia membuat mesin pencari harus semakin selektif dalam proses *indexing* sebuah halaman situs, maka mesin pencari membuat standarisasi dari konten situs *web*. Maka memerlukan waktu yang cukup lama untuk mendapatkan pengunjung pada situs *web* yang direkomendasikan oleh mesin pencari sesuai sesuai dengan isi konten situs *web*.

II.2.13.5 Strategi Penerapan SEO

Strategi penerapan SEO merupakan teknik yang paling efektif, untuk memaksimalkan kinerja situs *web hosting* sehingga situs *website* yang menggunakan teknik ini akan lebih maksimal, berikut stategi yang diterapkan.

a. Usia *Domain*

   Penerapan teknik SEO ini akan berjalan dengan baik jika usia *domain* sudah memiliki usia minimal 6 bulan dari tanggal registrasi kepemilikan nama *domain*.

b. Uniq Page Title

   Page Title adalah pemberian judul halaman *web* yang akan menjadi informasi pertama yang akan direkam oleh mesin, sehingga bisa dimengerti oleh pengunjung situs *website*.

c. *Meta Description*

Meta Description adalah *tag* untuk memberikan penjelasan mengenai halaman tersebut, ini akan menjadi info penting yang akan disimpan oleh mesin pencari yang akan diberikan pada *user* yang mencari sesuai dengan kategori yang dimaksud.

d. URL

URL adalah alamat sebuah halaman *web*site yang akan memudahkan mensin pencari untuk membaca halaman yang sesui dengan kata kunci yang dimaksud.

e. Site Navigation

Site Navigation adalah untuk memberikan petunjuk kepada pengunjung bahwa alur *link* di dalam situs tersebut.

f. Anchot Text

Anchot Text berfungsi untuk memberikan tanda bahwa kategori tersebut berhubungan dengan halaman yang lainnya.

g. Tags Appropriatly

Tags Appropriatly berfungsi untuk menampilkan struktur halaman kepada pengguna.

h. Images

Images berfungsi untuk memudahkan pengunjung jika suatu *file* gambar tidak bisa diakses maka akan ada keterangan mengenai gambar tersebut.

i. Robots

Robots berfungsi untuk memberitahukan pada robot google bahwa halaman atau direktori mana saja yang tidak boleh diakses [8].

II.2.14 *Backup* dan *Restore Database*

Pada teknologi informasi *backup* mengacu pada menyalin data sedangkan *restore* merupakan mengembalikan data, dimana data tersebut merupakan data salinan yang dapat di*restore* (mengembalikan) apabila ada data yang hilang data salinan tersebut biasa disebut dengan *backup*. *Backup* berguna untuk dua tujuan utama, pertama untuk *restore* (mengembalikan) data yang mengalami kerusakan akibat bencana alam (misalkan banjir, gempa bumi dan lain-lain), kedua untuk mengembalikan *file* setelah mengalami kesalahan menghapus atau korupsi. Para pemakai memahami bahwa korupsi akan terjadi, *hard drives* akan gagal, *motherboards* rusak, dan kekeliruan dalam menghapus data maka diperlukan *backup* data. *File* yang merupakan salinan dari *file* yang masih aktif dalam *database* sebagai pelindung atau cadangan bila *file database* rusak atau hilang disebut *backup file* [12].

II.2.15 *Flowmap*

Bagian alir atau *flowmap* adalah bagan yang menunjukkan alir di dalam program atau prosedur sistem secara logika. Bagian alir digunakan terutama untuk alat bantu komunikasi dan dokumentasi. Bagian alir dokumen atau sering disebut juga *flowmap* merupakan bagan alir yang menunjukkan arus dari aliran formulir termasuk tembusannya [2].

II.2.16 ERD (*Entity Relational Diagram*)

Model E-R didasarkan pada persepsi bahwa dunia nyata merupakan sekumpulan dari sejumlah objek dasar (entitas) dan relasi antar objek-objek data tersebut. Diagram yang menggambarkan struktur *logic* keseluruhan basis data, simbol yang digunakan adalah persegi empat, merepresentasikan himpunan entitas (untuk entitas lemah diberi garis ganda), ellips, merepresentasikan atribut, wajik, merepresentasikan himpunan keterhubungan, garis, menghubungkan simbol-simbol pada diagram. Label dari persegi empat, ellips, dan wajik menunjukkan nama, kardinalitas pemetaan dinyatakan dengan 2 (dua) cara yaitu *Korth* garis berarah (1) dan garis tidak berarah (banyak), *date* menuliskan kardinalitasnya pada garis dan peran dapat dituliskan sebagai label dari garis [3].

Pemakaian elemen-elemen dalam ERD ada tiga diantaranya adalah sebagai berikut:

1. *Entity* (entitas) adalah sebuah objek yang dapat dibedakan dari objek-objek lainnya, yang memiliki sejumlah properti atau atribut, dimana setiap atribut memiliki sekumpulan nilai yang diijinkan yang disebut domain, himpunan entitas yaitu kumpulan jumlah entitas yang memiliki tipe yang sama dan sebuah basis data mengandung sekumpulan himpunan entitas yang masing-masingnya memiliki sejumlah entitas dari tipe yang sama.

   Entitas terdiri dari 2 (dua) macam yaitu :

   - *Strong entity* (entitas kuat) : entitas yang mandiri, yang keberadaannya tidak bergantung pada keberadaan entitas yang lainnya. Instansiasi entitas kuat selalu memiliki karakteristik yang unik disebut *identifier* (sebuah atribut

tunggal atau gabungan atribut-atribut yang secara unik dapat digunakan untuk membedakannya dari entitas kuat yang lain).

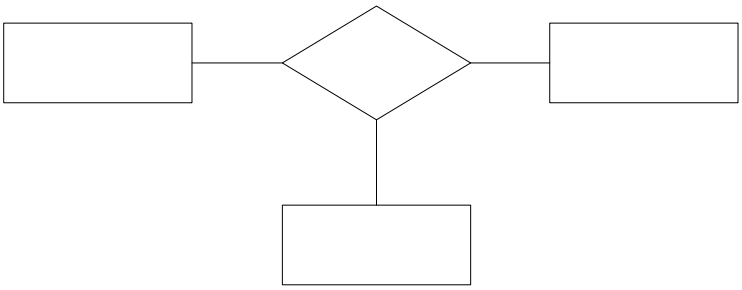
- *Weak entity* (entitas lemah) : entitas yang keberadaannya sangat bergantung pada keberadaan entitas yang lainnya. Entitas lemah tidak memiliki arti apa-apa dan tidak dikehendaki kehadirannya dalam diagram ER tanpa kehadiran entitas di mana mereka bergantung.

2. *Relationship* (relasi) merupakan hubungan antar entitas yaitu sebuah relasi menggambarkan suatu asosiasi antar sejumlah entitas, himpunan relasi (*relationship set)* adalah kumpulan sejumlah relasi yang memiliki tipe yang sama yang merupakan relasi matematis terhadap dua atau lebih himpunan entitas : {e1, e2,……,en) | (e1 ∈ E1, e2 ∈ E2, ……en ∈ En)}, Jumlah entitas terlihat dalam 2 (dua) buah relasi disebut derajat. Kebanyakan relasi yang muncul adalah relasi *binary*, ada beberapa yang *ternary*, lebih dari itu sangat jarang, fungsi sebuah entitas didalam relasi disebut peran (*role*) dan sebuah relasi dapat memiliki atribut.

3. Agregasi adalah suatu keadaan di mana suatu relasi hanya dapat direalisasikan setelah relasi yang lain ada terlebih dahulu. Relasi yang bertipe agregasi tidak dimungkinkan ada jika relasi yang menjadi prasyaratnya tidak terealisasi. Agregasi dapat dipandang sebagaimana relasi pada umumnya (yang menghubungkan 2 entitas). Karena relasi ini dibentuk dari relasi lain (relasi prasyarat) yang secara kronologis lebih dulu terbentuk, maka pengimplementasiannya juga harus dilakukan setelah relasi prasyarat tersebut

terimplementasikan. Selanjutnya kita tinggal meninjau derajat relasi dari relasi agregasinya.

4. *Ternary Degree* (Derajat Tingkat Tiga) merupakan Satu *relationship* menghubungkan 3 (tiga) *entity* yang berbeda satu sama lain sehingga bisa menghasilkan tabel baru. Gambar *ternary degree* dapat dilihat pada gambar II.9.



Gambar II.9 *Ternary Degree*

II.2.17 Diagram Konteks

Diagram konteks adalah diagram tingkat atas, dan merupakan diagram dari sebuah sistem yang menggambarkan aliran-aliran data yang masuk dan keluar dari sistem dan yang masuk dan keluar dari *entitas* [1].

Diagram konteks menyoroti sejumlah karakteristik penting sistem, yaitu :

1. Kelompok, pemakai, organisasi atau sistem lain dimana sistem melakukan komunikasi (sebagai *terminator*).
2. Data masuk, yaitu data yang diterima sistem dari lingkungan dan harus diproses dengan cara tertentu.
3. Data keluar, yaitu data yang dihasilkan sistem dan diberikan ke dunia luar.

4. Penyimpanan data (*storage)*, yaitu digunakan secara bersamaan antara sistem dengan *terminator*. Data ini dibuat oleh sistem dan digunakan oleh lingkungan atau sebaliknya. Hal ini berarti pembuatan simbol *data storage* dalam diagram konteks dibenarkan, dengan syarat simbol tersebut merupakan bagian dari dunia di luar sistem.
5. Batasan, antara sistem dan lingkungan.

   Simbol yang digunakan dalam diagram konteks, antara lain :

   a. Persegi panjang**,** untuk berkomunikasi langsung dengan sistem melalui aliran data.
   b. Lingkaran**,** untuk menunjukkan adanya kegiatan proses dalam sistem.

II.2.18 DFD (*Data Flow Diagram*)

DFD sering digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut akan disimpan. DFD merupakan alat yang digunakan pada metodologi pengembangan sistem yang terstruktur [1]. Sedangkan arti dari DFD tersebut adalah merupakan alat perancangan sistem yang berorientasi pada alur data dengan konsep dekomposisi dapat digunakan untuk penggambaran analisa maupun rancangan sistem yang mudah dikomunikasikan oleh profesional sistem kepada pemakai maupun pembuat program, ataupun dapat diartikan sebagai penggambaran arus data dari suatu sistem informasi, baik sistem lama maupun sistem baru secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut berada.

Diagram ini menjelaskan bagaimana data masukan diubah menjadi keluaran, dimana setiap bagian pada diagram menjelaskan proses transformasi yang berbeda. DFD dapat digunakan untuk menggambarkan sistem atau perangkat lunak pada banyak tingkatan dari suatu abstraksi. DFD dapt dibagi kedalam tingkatan-tingkatan yang menggambarkan bertambahnya atau meningkatkan aliran data dan rincian fungsional. Level-0 pada DFD, menyatakan prosedur-prosedur dari keseluruhan sistem yang digambarkan pada model konteks. Sedangkan model konteks merupakan model sistem pokok yang menggambarkan keseluruhan elemen perangkat lunak sebagai sebuah lingkaran dengan data masukan dan data keluaran dinyatakan dengan anak panah masuk atau keluar berturut-turut. Dengan kata lain model konteks menjelaskan bagaimana gambaran sistem dan hubungan dengan pihak luar.

Komponen-komponen yang ada dalam DFD terdiri dari 4 (empat) macam yaitu *External entity* (kesatuan luar) atau *boundary* (batas sistem), *data flow* (Arus data), *process* (proses), *data store* (penyimpanan data).

a. *External entity* (kesatuan luar) atau *boundary* (batas sistem)

   Setiap sistem pasti memiliki batas sistem *(boundary)* yang memisahkan suatu sistem dengan lingkungan luarnya. Kesatuan luar *(external entity)* merupakan kesatuan di lingkungan luar sistem yang dapat berupa orang, organisasi atau sistem lainya yang berada di lingkungan luarnya yang memberikan *input* atau menerima *output* dari sistem.

b. *Data flow* (arus data)

Arus data di DFD diberi simbol panah. Arus data ini mengalir diantara proses, simpanan, dan kesatuan luar.

c. *Process* (proses)

Suatu proses adalah kegiatan atau kerja yang dilakukan oleh orang, mesin atau komputer dari hasil suatu arus data yang masuk ke dalam proses untuk dihasilkan arus data yang akan keluar dari proses.

d. *Data store* (penyimpanan data)

Penyimpanan data *(data store)* merupakan simpanan dari data yang dapat berupa suatu *file* atau *database* di komputer, suatu arsip atau catatan manual dan lain sebagainya [1].