BAB 2

TINJAUAN PUSTAKA

2.1 Tinjauan Perusahaan

Berikut akan dijelaskan mengenai sej,arah Toko Nuy Collection's, visi dan misi, struktur organisasi, serta deskripsi tugas.

2.1.1 Sejarah Toko Nuy Collection's

Toko Nuy Collection's merupakan toko yang bergerak dalam bidang penjualan produk *fashion* muslimah. Toko Nuy Collection's didirikan pada tahun 2011 oleh Ibu Nurhayati. Toko ini berlokasi di jalan raya Margasari No. 195 Kota Bandung.

2.1.2 Visi dan Misi

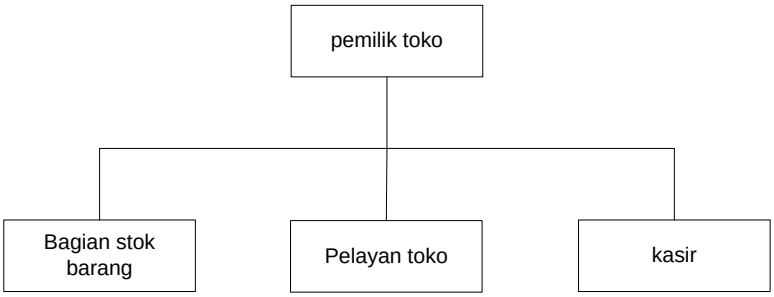
Visi dari toko Nuy Collection's adalah untuk mengembangkan usaha atau bisnis dalam bidang penjualan barang-barang kebutuhan para muslimah, juga menguntungkan dalam industri perdagangan dan dapat meningkatkan kesejahteraan orang banyak.

Adapun misi dari toko Nuy Collection's adalah senantiasa berusaha memberikan pelayanan terbaik bagi seluruh konsumen toko Nuy Collection's, bahkan melebihi harapan konsumen sekaligus memberikan manfaat bagi semua kalangan masyarakat

2.1.3 Struktur Organisasi

Struktur organisasi merupakan gambaran formal organisasi yang menunjukkan adanya pemisahan fungsi, uraian tugas, wewenang dan tanggung jawab yang di susun untuk membantu pencapaian hasil usaha perusahaan serta pengendaliannya secara efektif. Bentuk organisasi Toko Nuy Collection's didasarkan atas tinjauan dari segi wewenang, tanggung jawab dan hubungan kerja dalam fungsi organisasi.

Toko Nuy Collection's yang termasuk ke dalam kategori perusahaan perseorangan memiliki struktur organisasi sederhana yang dapat di lihat pada gambar 2.1.



Gambar 2.1 Struktur Organisasi Toko Nuy Collection's

2.1.4 Deskripsi Tugas

Definisi tugas digunakan untuk mengetahui tugas, wewenang, tanggung jawab dari masing-masing bagian. Definisi tugas yang ada di toko Nuy Collection's adalah sebagai berikut:

1. Pemilik Toko

   Pemilik toko Nuy Collection's memiliki tugas sebagai pengambil keputusan, mengontrol kinerja semua pegawai, melaksanakan pemeriksaan

yang meliputi seluruh aspek kegiatan manajemen keuangan dan operasional agar pengelolaan dapat dilaksanakan secara efektif dan efisien.

2. Bagian stok barang

   Bagian stok barang di toko Nuy Collection's bertugas memeriksa persediaan barang, dan melakukan pemesanan barang ke *supplier* jika persediaan barang habis.

3. Pelayan Toko

   Pelayan toko di toko Nuy Collections bertugas melayani konsumen yang datang langsung ke toko.

4. Kasir

   Kasir bertugas untuk menghitung uang hasil penjualan dan bertanggung jawab atas uang yang masuk, melaporkan hasil penjualan harian.

2.2 Landasan Teori

Berikut akan dijelaskan mengenai e-commerce, pengenalan CSS, pengenalan PHP dan HTML.

2.2.1 E-Commerce

Berikut akan dijelaskan mengenai sejarah e-commerce, keuntungan e-commerce, kerugian e-commerce dan klasifikasi e-commerce.

2.2.1.1 Sejarah E-commerce

Saat awal ditemukannya jaringan komputer, kebanyakan perusahaan bisnis skala besar di seluruh dunia, terutama di Amerika Serikat, menggunakan suatu

bagian tertentu dari perdagangan elektronik (*electronic commerce)* untuk mengendalikan transaksi antarbisnis. EDI (*Electronic Data Interchange)*, yang memungkinkan pertukaran dokumen antarbagian dalam suatu perusahaan dengan bentuk yang terstandarisasi di jaringan pribadi, telah di mulai pada sekitar tahun 1960-an di Amerika Serikat. Kemudian aplikasi-aplikasi perbankan berskala besar telah lama menggunakan jaringan terdedikasi (*dedicated networks)* untuk metode-metode pertransferan dana dengan menggunakan sistem EFT (*Electronic Fund Transfer)*, yang merupakan metode pentransferan dana secara elektronik, yang dirancang untuk mengoptimalkan pembayaran yang dilakukan secara elektronik.
Saat ini, dengan semakin maraknya penggunaan Internet, perdagangan secara elektronik (*e-commerce)* dilakukan oleh bisnis-bisnis dengan berbagai ukuran[3].

2.2.1.2 Keuntungan E-commerce

Adapun keuntungan e-commerce adalah sebagai berikut.

2.2.1.2.1 Keuntungan Bagi Perusahaan

Secara umum, ada berbagai manfaat yang dapat didapat perusahaan saat melakukan perdagangan secara elektronik. Beberapa manfaat itu adalah sebagai berikut :

a. Memperpendek Jarak

   Perusahaan-perusahaan dapat lebih mendekatkan diri dengan konsumen. Dengan hanya mengklik *link-link* yang ada pada situs-situs, konsumen dapat menuju ke perusahaan dimanapun saat itu konsumen berada.

b. Perluasan Pasar

Jangkauan pemasaran menjadi semakin luas dan tidak terbatas oleh area geografis dimana perusaan berada.

c. Perluasan Jaringan Mitra Bisnis

Pada perdagangan tradisional, sangat sulit bagi suatu perusahaan untuk mengetahui posisi geografis mitra kerjanya yang berada di negara-negara lain atau benua lain. Bagaimana pun juga mitra kerja sangat penting untuk konsultasi dan kerjasama baik teknis maupun non-teknis. Dengan adanya perdagangan elektronik lewat jaringan internet, hal-hal tersebut bukan menjadi masalah yang besar lagi.

d. Efisien

Seperti di singgung di bagian sebelumnya, perdagangan elektronik akan sangat memangkas biaya-biaya operasional. Perusahaan-perusahaan yang berdagang secara elektronik tidak membutuhkan kantor dan toko yang besar, menghemat kertas-kertas yang digunakan untuk transaksi-transaksi, periklanan, serta pencatatan-pencatatan. Selain itu, perdagangan elektronik juga sangat efisien dari sudut waktu yang digunakan. Pencarian informasi-informasi produk/jasa dan transaksi - transaksi bisa dilakukan lebih cepat serta lebih akurat.

2.2.1.2.2 Keuntungan Bagi Konsumen

Perdagangan elektonik juga bermanfaat bagi pelanggan/konsumen. Beberapa manfaat itu adalah sebagai berikut:

a) Efektif

Konsumen dapat memperoleh informasi tentang produk/jasa yang dibutuhkannya dan bertransaksi dengan cara yang cepat dan murah.

b) Aman Secara Fisik

Konsumen tidak perlu mendatangi toko tempat perusahaan menjajakan harganya dan ini memungkinkan konsumen dapat bertransaksi dengan aman sebab di daerah-daerah tertentu mungkin sangat berbahaya jika berkendaraan dan membawa uang tunai dalam jumlah yang besar.

c) Fleksibel

Konsumen dapat melakukan transaksi dari berbagai lokasi, baik dari rumah, kantor, warnet, atau tempat-tempat lainnya. Konsumen juga tidak perlu berdandan rapi seperti pada perdagangan tradisional umumnya.

2.2.1.2.1 Keuntungan Bagi Masyarakat Umum

Perdagangan elektonik juga bermanfaat bagi masyarakat umum. Beberapa manfaat itu adalah sebagai berikut:

1. Mengurangi Polusi dan Pencemaran Lingkungan

Dengan adanya perdagangan elektronik yang dapat dilakukan dimana saja, konsumen tidak perlu melakukan perjalanan - perjalanan ke toko-toko, dimana hal ini pada gilirannya akan mengurangi jumlah kendaraan yang berlalu - lalang di jalanan. Berkurangnnya kendaraan di jalanan berarti menghemat bahan bakar (BBM) dan mengurangi tingkat polusi udara sebab gas-gas buangan kendaraan bermotor dapat mencemari lingkungan.

2. Membuka Peluang Kerja Baru

Era perdagangan elektronik akan membuka peluang-peluang kerja baru bagi yang tidak ‘buta’ teknologi. Muncul pekerjaan-pekerjaan baru seperti pemrogram komputer, perancang web, ahli di bidang basis data, analisis sistem, ahli di bidang jaringan computer, dan sebagainya.

3. Menguntungkan Dunia Akademis

Berubahnya pola hidup masyarakat dengan hadirnya perdagangan elektronik, kalangan akademis akan semakin diperkaya dengan kajian-kajian psikologis, antropologis, sosial-budaya, dan sebagainya, yang berkaitan dengan cara dan pola hidup yang berkaitan dengan dunia maya. Selain itu, dampak langsung dari hadirnya Internet secara langsung akan menantang kiprah ilmuwan di bidang teknik komputer, teknik telekomunikasi, elektronika, pengembangan perangkat lunak, dan sebagainya.

4. Meningkatkan Kualitas Sumber Daya Manusia

Perdagangan elektronik, seperti juga teknologi komputer pada umumnya, hanya bias dilakukan oleh orang-orang yang tidak gaptek (gagap teknologi), sehingga pada gilirannya akan merangsang orang-orang untuk mempelajari teknologi komputer demi kepentingan mereka sendiri. Selain itu, dalam melakukan perdagangan elektronik, seseorang suatu saat mungkin akan ‘tersesat’ ke situs-situs berkualitas yang akan meningkatkan pemahaman orang yang bersangkutan.

2.2.1.3 Kerugian E-commerce

Di samping segala hal yang menguntungkan di atas, segala sesuatu ciptaan manusia pasti memiliki sisi negatif. Namun, dari sudut pandang manapun, perdagangan elektronik memiliki segi positif lebih banyak dari sisi negatifnya.

Sebagai langkah antisipasi, perlu memahami beberapa segi negatif perdagangan elektronik (atau internet pada umumnya) sebagai berikut:

a. Meningkatkan Individualisme

   Pada perdagangan elektronik, seseorang dapat bertransaksi dan mendapatkan barang/jasa yang diperlukannya tanpa perlu bertemu. Ini membuat beberapa orang menjadi berpusat pada diri sendiri (egois) serta individualistis dan merasa dirinya tidak terlalu membutuhkan kehadiran orang lain dalam hidupnya.

b. Terkadang Menimbulkan Kekecewaan

   Apa yang di lihat di layar monitor komputer kadang berbeda dengan apa yang di lihat secara kasat mata. Seseorang yang membeli lukisan di Internet mungkin suatu saat akan mendapati lukisannya tidak memiliki warna yang sama dengan apa yang dilihatnya di layar monitor. Seseorang yang membeli sofa di Internet adalah contoh lain. Di layar monitor sofa yang akan dibelinya terlihat begitu nyaman diduduki. Kenyataannya? Sangat mungkin, apa yang terlihat begitu lembut di layar monitor ternyata pada kenyataanya tidak begitu adanya.

c. Tidak Manusiawi

Sering sekali orang pergi ke toko-toko dan pusat-pusat perbelanjaan (mall) tidak sekedar ingin memuaskan kebutuhannya akan barang/jasa tertentu. Mungkin orang itu melakukannya untuk penyegaran (*refresing)* atau bersosialisasi dengan rekan-rekan atau keluarganya. Perdagangan elektronik gagal di pandang dari sudut pandang seperti ini. Di internet, meski dapat mengobrol (*chatting)* dengan orang lain, mungkin orang itu tidak merasakan jabat tangannya, senyuman ramahnya, atau candanya.

Setelah melihat bagaimana perdagangan elektronik yang dilakukan secara seksama akan sangat menguntungkan baik perusahaan penjual, pembeli, maupun masyarakat pada umumnya. Meski demikian, dengan efek-efek negatifnya, perdagangan elektronik seharusnya tidak menggantikan perdagangan tradisional. Perdagangan tradisional menawarkan komunikasi, keakraban, kehangatan, dan sebagainya, dimana hal ini tidak dapat di jumpai di perdagangan elektronik. Perdagangan elektronik sangat berorientasi transaksi. Perdagangan elektronik hanya sesuai pada situasi dan kondisi tertentu, dimana kontak langsung antara penjual dan pembeli tidak terlalu dibutuhkan.

2.2.1.4 Klasifikasi E-Commerce

Penggolongan *E-Commerce* yang lazim dilakukan orang ialah berdasarkan sifat transaksinya, antara lain:

1. *Business to Business* (B2B)

   Jenis transaksi di mana pembeli biasanya membeli dalam jumlah besar karena akan di jual kembali. Contoh penjualan grosir.

2. *Business to Consumer* (B2C)

   Jenis transaksi dimana pembelinya perorangan dan tidak punya tujuan untuk menjualnya kembali biasanya semacam toko *online* yang menjual berbagai macam barang.

3. *Consumer to Consumer* (C2C)

   Jenis transaksi di mana pembelinya perorangan yang tidak mempunyai tujuan untuk di jual kembali dan penjualnya juga perorangan yang tidak menyediakan bermacam-macam barang melainkan hanya beberapa barang saja. Contoh: *online advertising*.

4. *Consumer to Business* (C2B)

   Termasuk ke dalam kategori ini adalah perseorangan yang menjual produk atau layanan kepada organisasi, dan perseorangan yang mencari penjual, berinteraksi dengan mereka dan menyepakati suatu transaksi.

2.2.2 Pengenalan CSS

CSS atau *Cascading Style Sheet* saat ini adalah sebuah elemen penting dalam pembuatan sebuah *web*. Sama pentingnya saat mendesain bentuk *web* pada Photoshop. Dengan CSS, dapat mendesain sejumlah halaman dengan aturan yang sama tanpa mengubah halaman-halaman tersebut satu persatu. Bayangkan saja jika mempunyai ratusan halaman dan harus mendesign ulang halaman tersebut satu persatu, hanya akan menghabiskan waktu. Dengan CSS, masalah tersebut

akan diatasi. Dengan membuat sebuah file CSS kemudian meng-importnya ke dalam *file* halaman *web*, sehingga dapat memformat *style* semua halam *web* yang di buat.

Saat ini CSS merupakan aturan *style* yang banyak digunakan dikarenakan fleksibilitas dari tag, kemudahan dan lengkapnya atribut yang dimilikinya.

Penggunaan CSS dalam *web* akan lebih efisien dikarenakan CSS dapat digunakan secara berulang pada tag-tag tertentu, sehingga tidak usah mengetikkan ulang seluruh perintah pemformatan desain seperti halnya dalam HTML klasik[1].

2.2.3 Pengenalan PHP

PHP atau *Hypertext Preprocessor* adalah salah satu jenis bahasa pemograman *web* yang *open source*, sehingga dapat digunakan secara cuma-cuma. Selain gratis, PHP juga memiliki kelebihan diantaranya, kemudahan dalam menggunakannya serta dapat digunakan untuk membuat *website* dinamis.

Dengan PHP dapat membuat situs lebih interaktif. Misalnya untuk pengisian formulir, pengisian buku tamu, dll. PHP juga dapat menghubungkan *website* yang di buat dengan database, sehingga menciptakan *website* yang dinamis.

Struktur dari file PHP biasanya didahului oleh tag <?php serta di tutup dengan tag ?>. Dan file-nya ber-ekstensi .php.

<?php

Disini letak skrip-skrip PHP

?>

Setiap perintah juga harus dilengkapi dengan titik koma “;”. Serta variable PHP di tulis dengan tanda dolar ($) dan diikuti nama variabel tanpa spasi.

```
// Perintah diakhiri dengan “;”
<?php
Perintah PHP1;
Perintah PHP2;
?>
// Variabel tanpa spasi
<?php
$variabelphp = “Nilai dari variabelphp”;
$tulis = “value dari tulis”;
?>
```

Untuk menghubungkan web berbasis PHP dengan database, biasanya akan membutuhkan sebuah perantara koneksi. Kondisi ini dapat dibuat dengan mudah menggunakan *tools Recordest* yang ada pada *Dreamweaver*. Tapi untuk sederhananya sebagai berikut:

```
<?php
$servername = “localhost”;
$username = “root”;
$password = “password”;
$database = “database”;

mysql_connect ($servername, $username, $password)
```

or die (“Gagal Koneksi”);

mysql_select_db($database) or die (“Database tidak ada”);

?>

Kemudian simpan file tersebut pada folder *root website*, maka setiap file yang perlu untuk mengakses *database* akan memerlukan *file* koneksi tersebut. Gunakan fungsi *require()* atau *include()* pada halaman PHP yang memerlukan *file* koneksi[1].

2.2.4 HTML (*Hypertext Markup Language*)

Hypertext markup Language (HTML) dalam ilmu komputer merupakan bahasa pemformatan teks untuk dokumen-dokumen pada jaringan komputer yang di kenal sebagai *World Wide Web* (atau sering disebut *web* saja). Dokumen-dokumen HTML merupakan berkas teks yang mengandung 2 bagian:

1. Isi yaitu segala sesuatu yang ingin ditampilkan dan diperlihatkan dalam dokumen web.
2. Tag yang merupakan informasi pemformatan, yang tersembunyi dari pandangan pengguna, yang memberitahu *browser* tentang bagaimana caranya menampilkan isi dokumen ke hadapan pengguna.

HTML pada dasarnya merupakan himpunan bagian dari bahasa yang lebih luas jangkauannya yaitu *Standard Generalized Markup Language* (SGML), yang merupakan sistem pengkodean dan pemformatan untuk dokumen yang ditampilkan di layar computer maupun di kertas.

Dokumen-dokumen SGML terdiri dari 3 jenis informasi: *data*, *structure*, dan *format*. Data mencakup tidak hanya teks, grafik dan gambar, suara tetapi juga

informasi-informasi yang tidak ditampilkan pada luaran misalnya basis data katalog pengiriman lewat surat yang memuat informasi rujukan tentang item tertentu yang berisi persediaan saat ini yang tidak tercantum dalam catalog tercetak.

Structure dokumen merujuk pada pengorganisasian data pada dokumen yang bersangkutan, misalnya bab-bab, subbab-subbab, daftar-daftar, gambar-gambar, judul-judul, indeks, dan daftar isi. Halaman-halaman *World Wide Web* memuat struktur yang sama. Namun selain itu, HTML juga memuat *link*, atau hubungan dengan halaman-halaman *World Wide Web* yang lainnya.

Format merujuk pada bagaimana dokumen akan di lihat oleh para pengguna. Sebagai contoh, teks mungkin akan dimiringkan (*italic),* ditebalkan (*bold),* diberi garis bawah *(underline),* dan sebagainya.

World Wide Web pertama kali dikembangkan oleh fisikawan dan ilmuwan computer Inggris yang bernama Timothy Berners-Lee pada salah satu proyek yang diselenggarakan oleh Pusat Riset Nuklir Eropa *(European Organization for Nuclear Research* (CERN)*)* di Geneva, Swiss. Berners-Lee menggabungkan beberapa ide yang ada saat itu ke suatu sistem tunggal yang memudahkan para fisikiawan menggunakan data-data dan mengomunikasikannya lewat internet. Yang juga penting, Berners-Lee menambahkan kemamapuan multimedia pada dokumen-dokumen web. Berners-Lee mulai bekerja dengan *hypertext* pada awal tahun 1980-an dan prototipe-nya diimplementasikan oleh CERN pada tahun 1989. Selanjutnya ide-ide Berners-Lee secara cepat menyebar ke universitas-universitas di seluruh dunia (dan akhirnya ke dunia komersial).

Beberapa *tag* dalam dokumen-dokumen *HTML* menentukan bagaimana teks-teks di format. *Tag-tag* yang lain memberitahu komputer tentang bagaimana menanggapi aksi-aksi yang datang dari pengguna. Sebagai contoh, apa yang harus dilakukan komputer saat pengguna mengklikkan mouse-nya pada icon tertentu. Kemudian *tag* lain yang penting adalah *link* yang mengandung *uniform Resource locator* (ULR), yang merujuk pada dokumen lain di *server* yang sama atau computer lain yang ada di jaringan global internet. Selain itu, HTML juga memiliki penandaan untuk membuat formulir-formulir yang memungkinkan pengguna mengisi informasi tertentu dan mengirimkannya secara elektronik ke tempat-tempat lain di jaringan internet.

Perangkat lunak yang memungkinkan pengguna menjelajahi *World Wide Web* dan melihat dokumen-dokumen HTML dinamakan *browser. Browser* menafsirkan *tag-tag* HTML pada dokumen dan melakukan pemformatan saat dokumen yang bersangkutan ditampilkan di layar monitor.

Pada tahun 1989, Berners-Lee merumuskan suatu proposal tentang sebuah system *hypertext* yang memiliki 3 komponen sebagai berikut:

1. Antarmuka yang konsisten untuk semua jenis *platform.* Antarmuka ini harus menyediakan akses yang dapat digunakan oleh berbagai jenis komputer.
2. Akses informasi yang bersifat *universal*. Setiap pengguna harus dapat mengakses setiap informasi yang berbeda.
3. Antarmuka yang menyediakan akses keberbagai jenis dokumen dan protokol.

HTML 2.0 didefinisikan oleh *Internet Engineering Task Force* (IETF) dengan himpunan fitur dasar, termasuk kemampuan halaman yang bersifat interaktif. Versi selanjutnya menambahkan fitur-fitur tambahan seperti teks yang berkedip *(blinking text),* latar belakang yang dapat di atur, serta table-tabel. Versi-versi selanjutnya menambahkan kemampuan-kemampuan baru dan setiap saat *browser* harus dimodifikasi untuk dapat menyesuaikan diri dengan kemampuan-kemampuan baru tersebut. Saat ini standar untuk HTML dispesifikasi oleh W3C (*world Wide Web Consortium)*[2].

2.2.5 JavaScript

JavaScript adalah bahasa skrip dan *Netscape* yang fungsinya mirip dengan *VBScript* namun bahasa dasarnya adalah adalah *Java*. *Javascript* di dukung oleh *browser* Netscape Navigator mulai versi 2.0. *Javascript* lebih mudah digunakan dari *Java*, tetapi tentu saja tidak memiliki kemampuan yang secanggih *Java*. *Javascript* menggunakan bahasa HTML sebagai antarmuka penggunanya. *Java* dapat membuat antarmuka secara lengkap. Pada sisi *client, applet* (sebuah aplikasi sikembangkan untuk dijalankan di *browser*) di pelihara dalam kode sumber. Pada *server*, mereka dikompilasimenjadi kode *byte* (bahasa antara), mirip dengan program-program yang dibuat dengan bahasa pemograman *Java*. *Javascript* berevolusi dari bahasa *livescript* dan *Netscape* dan kompatibel dengan bahasa pemograman *Java*. *Javascript* tidak sehebat *Java*, namun dapat digunakan bersama dengan *Java*. Sebagai contoh *applet Javascript* dapat digunakan untuk membuat form masukan pengguna dan melakukan validasi asupan, sementara program *Java* akan melakukan pemrosesan informasinya[7].

2.2.5.1 Kekurangan dan Kelebihan JavaScript

Beberapa kelebihan dari *JavaScript* diantaranya adalah sebagai berikut:

1. Ukuran file relatif kecil

   Kode *JavaScript* memiliki ukuran yang kecil sehingga ketika *web* yang mengandung *JavaScript* ditampilkan di *browser,* maka akses tampilannya akan lebih cepat dibandingkan ketika membuka suatu *web* yang mengandung kode *Java*. Hal ini juga sangat berpengaruh dengan kinerja *server*. Semakin kecil *space* suatu *web* yang di simpan dalam suatu *server,* maka kinerja *server* ketika di-*browsing* oleh *user* di internet akan tidak terlalu berat, selain itu sifat *JavaScript* yang *client side* tidak perlu lagi diolah oleh *server* ketika *browser* memanggilnya.

2. Mudah untuk dipelajari

   Javascript merupakan bahasa semi pemograman yang merupakan gabungan antara bahasa pemograman *Java* dengan bahasa kode HTML sehingga di sebut bahasa *hybrid,* dan tidak memiliki aturan serumit *Java.*

3. Terbuka

   Javascript tidak terikat oleh *hardware* maupun *software* tertentu, bahkan sistem operasi seperti *Windows* maupun *Unix.* Karena bersifat terbuka, maka dapat dibuat di baca di semua jenis computer.

Sedangkan, beberapa kekurangan *JavaScript* diantaranya adalah sebagai berikut:

1. Skrip tidak terenkripsi

   Karena bersifat *client server*, maka kode yang di buat di *text editor* dan telah dijadikan semua halaman *web* di *server*, ketika *user* me-*request* halaman tersebut, maka kode *javaScricpt* akan langsung ditampilkan di *browser*. *User* bisa melihat dan meniru dari *source*-nya secara langsung tanpa enkripsi sedikitpun.

2. Kemampuan terbatas

   Walaupun *JavaScript* mampu membuat halaman *web* menjadi interaktif dan dinamis, namun *JavaScript* tidak mampu membuat program aplikasi sendiri seperti *Java*.

3. Keterbatasn objek

   Javascript tidak mampu membuat kelas-kelas yang bisa menampung objek-objek tambahan seperti *Java*, karena *JavaScript* telah memiliki objek *built-in* pada struktur bahasanya[7].

2.2.6 Mengenal MySQL dan phpMyAdmin

MySql adalah *database* yang menghubungkan *script* PHP menggunakan perintah *query* dan *escape character* yang sama dengan PHP. *MySQL* mempunyai tampilan *client* yang mempermudah dalam mengakses *database* dengan kata sandi untuk mengijinkan proses yang boleh dilakukan.

Sedangkan *phpMyAdmin* merupakan halaman yang terdapat pada *web server*. Fungsi dari halaman ini adalah sebagai pengendali *database MySQL* menggunakan *web server*[1].

2.2.6.1 Perintah Membuat Database

Untuk membuat *database* badu di dalam *server* dengan menggunakan printah *CREATE DATABASE*. Berikut adalah contoh untuk membuat *database*.

mysql> CREATE DATABASE nama database;

Query OK, 1 row affected (0.28)

Pada perintah di atas, pernyataan *"Query OK, 1 row affected (0.02)"* menendakan bahwa perintah yang dijalankan dapat diterima oleh *server*. Sehingga maksud dari perintah akan diterapkan pada *database MySQL*.

Untuk melihat hasilnya, dapat menggunakan perintah SHOW seperti sebelumnya.

mysql> SHOW DATABASES;

2.2.6.2 Perintah Mengaktifkan Database

Setelah *database* di buat, maka dapat bekerja di dalam *database* tersebut dengan menggunakan perintah-perintah DML (*Data Manipulation Language*) dan DDL (*Data Definition Language*). Untuk dapat bekerja pada sebuah *database*, masuklah ke dalam *database* tersebut. Caranya adalah dengan menggunakan perintah *USE*. Perhatikan contohnya:

mysql> USE nama database;

Database changed

Setelah mendapatkan pesan "*Database changed*", maka sekarang berada di dalam *database*. Di dalam *database* tersebut dapat menggunakan berbagai operasi DML dan DDL.

2.2.6.3 Perintah Membuat Tabel

Di dalam *databse*, dapat melakukan operasi DDL seperti perintah *CREATE* untuk membuat table. Struktur perintah untuk membuat table adalah sebagai berikut:

Sintaks

CREATE TABLE namatabel (koloma TYPE(panjang),

Kolomb TYPE(panjang),

……,

Kolomn TYPE(panjang));

2.2.6.4 Perintah Menampilkan Tabel

Melihat kondisi daftar table di dalam *database* yang sedang aktif. Perintah yang dapat digunakan adalah *SHOW TABLES*. Perhatikan contoh *query*-nya sebagai berikut:

mysql> SHOW TABLES;

2.2.6.5 Perintah mendeskripsikan Tabel

Semua desain kolom tabel yang telah di buat, dapat ditampilkan lagi dengan menggunakan perintah *DESCRIBE* atau *DESC*. Perhatikan caranya sebagai berikut:

mysql> DESCRIBE namatabel;

2.2.6.6 Perintah Memasukkan Data Pada Tabel

Setelah tabel terbentuk, dapat menggunakan perintah-perintah (*Data Manipulation Language*) seperti *INSERT*, *UPDATE*, dan *DELETE* untuk memanipulasi data yang ada didalamnya.

Perintah *INSERT* digunakan untuk memasukkan data baru ke dalam tabel, cara menggunakannya adalah sebagai berikut:

Sintaks

INSERT INTO namatabel (koloma, kolomb, ..., kolomn)

Values ('datakoloma', 'datakolomb', '...', 'datakolomn')

2.2.6.7 Perintah Menampilkan Isi Tabel

Setelah data dimasukkan, dapat melihat daftar data yang ada di dalam tabel dengan nama tabel yang telah di buat. Caranya adalah dengan menggunakan perintah *SELECT*, perhatikan contohnya sebagai berikut:

*mysql> SELECT * FROM namatabel;*

2.2.6.8 Menampilkan Dengan Kondisi

Menampilkan data per kolom atau banyak kolom yang telah dikuasai, permasalahannya sekarang adalah bagaimana apabila hendak memilih baris data tertentu saja untuk di tampilkan. Apabila memiliki keinginan tersebut, maka dapat menggunakan parameter *WHERE*.

Parameter *WHERE* merupakan bentuk klausa *SQL* standar yang dimiliki oleh *MySQL* dalam membuat pengkondisian dalam permintaan/*query*. Sintaks penggunaannya adalah:

Sintaks dasar

SELECT [| kolom] FROM namatabel WHERE kondisi;*

2.2.6.9 Merubah Data dengan Kondisi

Perintah DML lain yang berkaiatan dengan data adalah *UPDATE.* Perintah ini dapat digunakan untuk meremajakan isi data lama pada atabel menjadi data terbaru. Penggunaan dari perintah tersebut dapat di lihat pada sintaks berikut:

Sintaks dasar

UPDATE namatabel SET koloma='databaru',

Kolomb='databaru' WHERE kondisi;

2.2.6.10 Menghapus Data dengan Kondisi

Dengan menggunakan parameter *WHERE*, dapat melakukan operasi penghapusan data dari tabel anggota. Perintah DML yang digunakan untuk menghapus data adalah *DELETE*. Berikut sintaksnya:

Sintaks dasar

DELETE FROM namatabel WHERE kondisi;

2.2.7 Basis Data (*Database*)

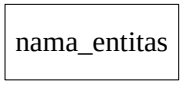
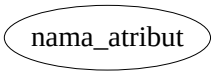
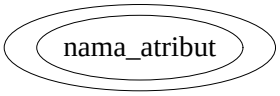
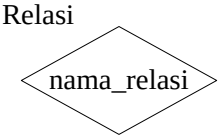
Definisi Basis Data (*Database*) dapat didefinisikan dalam sejumlah sudut pandang seperti:

a. Sistem yang terkomputerisasi yang tujuan utamanya adalah memelihara informasi dan membuat informasi tersebut tersedia saat dibutuhkan.
b. Kumpulan data yang saling berhubungan yang di simpan di tempat penyimpanan dan dapat dimanfaatkan kembali dengan cepat dan mudah.

Sebuah basis data memerlukan system untuk memasukkan mengubah, menghapus, memanipulasi, dan dapat memperoleh data/informasi dengan praktis dan efisien, sistem tersebut dikenal dengan nama *Database Management System* (DBMS). DBMS mengatur proses pengambilan dan penyimpanan data seperti system operasi yang bertugas mengatur sumber daya di dalam komputer. DBMS akan menentukan bagaimana data di organisasi, di simpan, di ubah, dan di ambil kembali. DBMS juga menerapkan mekanisme pengamatan data, pemakaian data secara bersama-sama oleh beberapa aplikasi, dan menjaga kehandalan data (integritas data).

2.2.8 ERD

Pemodelan awal basis data yang paling banyak digunakan adalah menggunakan *Entity Relationship Diagram* (ERD). ERD dikembangkan berdasarkan teori himpunan dalam bidang matematika. ERD digunakan untuk pemodelan basis data relasional. Sehingga jika penyimpanan basis data menggunakan OODBMS maka perancangan basis data tidak perlu menggunakan ERD. Berikut adalah simbol-simbol yang digunakan pada ERD:

Simbol	Deskripsi
Entitas / *entity* 	Entitas merupakan data inti yang akan di simpan; bakal tabel pada basis data
Atribut 	*Field* atau kolom data yang butuh di simpan dalam suatu entitas
Atribut kunci primer 	*Field* atau kolom data yang butuh di simpan dalam suatu entitas dan digunakan sebagai kunsinakses *record* yang diinginkan; biasanya berupa id
Atribut multinilai / *multivalue* 	*Field* atau kolom data yang butuh di simpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu
Relasi 	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja
Asosiasi / *association* 	Penghubung antara relasi dan entitas di mana di kedua ujungnya memiliki *multiplicity* kemungkinan jumlah pemakaian

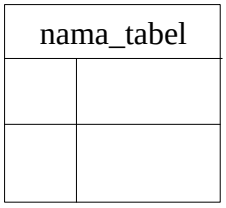
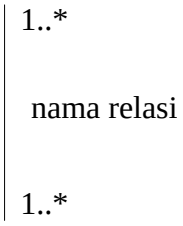
Gambar 2.2 Simbol ERD

2.2.9 CDM

CDM *(Conceptual Data Model)* atau model konsep data merupakan konsep yang berkaitan dengan pandangan pemakai terhadap data yang di simpan dalam basis data. CDM sudah di buat dalam bentuk tabel-tabel tanpa tipe data yang menggambarkan relasi antar tabel untuk keperluan implementasi ke basis

data.CDM merupakan hasil penjabaran labih lanjut dari ERD. Ada aturan-aturan yang harus diikuti dalam melakukan konversi ERD menjadi CDM.

Berikut adalah simbol-simbol yang ada pada CDM:

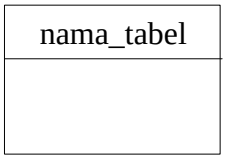
Simbol	Deskripsi
Entitas / tabel 	Entitas atau tabel yang menyimpan data dalam basis data
Relasi 	Relasi antar tabel yang terdiri atas nama relasi dan *multiplicity*

Gambar 2.3 Simbol CDM

2.2.10 PDM

Model Relasional atau *Physical Data Model* (PDM) adalah model yang menggunakan sejumlah tabel untuk menggambarkan data serta hubungan antara data-data tersebut. Setiap tabel mempunyai sejumlah kolom di mana setipa kolom memiliki nama yang unik beserta tipe datanya. PDM merupakan konsep yang menerangkan detail dari bagaimana data di simpan di dalam basis data.

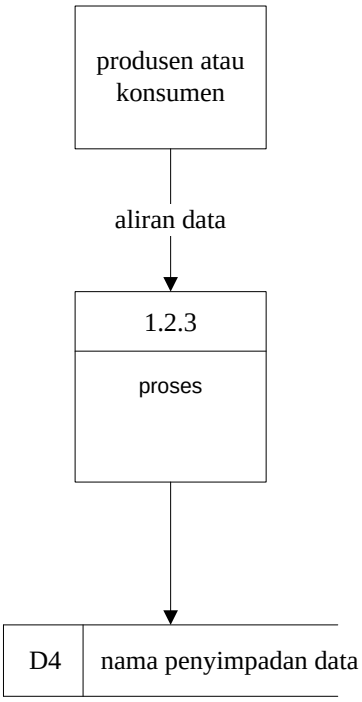
Berikut adalah simbol-simbol yang ada pada PDM:

Simbol	Deskripsi
Tabel  	Tabel yang menyimpan data dalam basis data
Relasi id_tbl 1 = id_fk_tbl2	Relasi antar tabel yang terdiri dari persamaan antara *primary key* (kunci primer) tabel yang di acu dengan kunci yang menjadi referensi acuan di tabel lain

Gambar 2.4 Simbol PDM

2.2.11 DFD

Data Flow Diagram (DFD) awalnya dikembangkan oleh Chris Gane dan Tris Saron pada tahun 1979 yang termasuk dalam *Structured System Analysis and Design Methodology* (SSADM) yang di tulis oleh Chris Gane dan Trish Saron. Sistem yang dikembangkan ini berbasis pada dekomposisi fungsional dari sebuah sistem. Berikut adalahcontoh DFDyang dikembangkan oleh Chris Gane dan Trish Sarson:



Gambar 2.5 Contoh DFD yang Dikembangkan Chris Gane & Trish Sarson

Edward Yourdon dan Tom DeMarco memperkenalkan metode yang lainpada tahun1980-an di mana mengubah persegi dengan sudut lengkung (pada DFD Chris Gane dan Trish Sarson) dengan lingkaran untuk menotasikan. DFD Edward Yourdon dan Tom DeMarco populer digunakan sebagai model analisis sistem perangkat lunak untuk sistem perangkat lunak untuk sistem perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur.

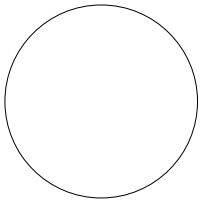
Informasi yang ada di dalam perangkat lunak dimodifikasi dengan beberapa transformasi yang dibutuhkan. *Data Floe Diagram* (DFD) atau dalam bahasa Indonesia menjadi Diagram Alir Data (DAD) adalah representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang

diaplikasikan sebagai data yang mengalir dari masukan *(input)* dan keluaran *(output).*

DFD dapat digunakan untuk merepresentasikan sebuah sistem atau perangkat lunak pada beberapa level abstraksi. DFD dapat di bagi menjadi beberapa level yang lebih detailuntuk merepresentasikan aliran informasi atau fungsi yang lebih detail. DFD menyediakan mekanisme untuk pemodelan fungsional ataupun pemodelan aliran informasi. Oleh karena itu, DFD lebih sesuai digunakan untuk memodelkan fungsi-fungsi perangkat lunak yang akan diimplementasikan menggunakan pemrograman terstruktur karena pemrograman terstruktur menbagi-bagi bagiannya dengan fungsi-fungsi dan prosedur-prosedur.

DFD tidak sesuai untuk memodelkan sistem perangkat lunak yang akan di bangun menggunakan pemrograman berorientasi objek. Paradigma pemrograman terstruktur dan pemrograman berorientasi objek merupakan hal yang berbeda. Jangan mencampuradukkan pemrograman terstruktur dan pemrograman berorientasi objek.

Notasi-notasi pada DFD (Edward Yourdon dan Tom DeMarco) adalah sebagai berikut:

Notasi	Keterangan
	proses atau fungsi atau prosedur; pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya menjadu fungsi atau prosedur di dalam kode program. catatan: nama yang diberikan pada sebuah proses biasanya berupa kata kerja
	*file* atau basis data atau penyimpanan *(storage)*; pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnyadibuat menjadi tabel-tabel basis data yang dibutuhkan, tabel-tabel ini juga harus sesuai dengan perancangan tabel-tabel pada basis data (*Entity Relationship Diagram* (ERD), *Conceptual Data Model* (CDM), *Physical Data Model* (PDM)) catatan: nama yang diberikan pada sebuah penyimpanan biasanya kata benda
	entitas luar *(external entity)* atau masukkan *(input)* atau keluaran *(output)* atau orang yang memakai/ berinteraksi dengan perangkat lunak yang dimodelakan atau sistem lain yang terkait dengan aliran data dari sistem yang di modelakan. catatan: nama yang digunakan pada masuan *(input)* atau keluaran *(output)* biasanya berupa kata benda
	aliran data; merupakan data yang di kirim antar proses, dari penyimpanan ke proses, atau dari proses ke masukkan (input) atau keluaran (output) catatan: nama yang digunakan pada aliran data biasanya berupa kata benda, dapat di awali dengan kata data misalnya ‘data siswa’ atau tanpa kata data misalnya ‘siswa’

Gambar 2.6 Notasi-notasi pada DFD

Berikut ini adalah tahapan-tahapan perancangan dengan menggunakan DFD:

1. Membuat DFD Lebel 0 atau sering disebut juga *Context Diagram* DFD Level 0 menggambarkan sistem yang akan di buat sebagai suatu entitas tunggal yang berinteraksi dengan orang maupun sistem lain. DFD Level 0 digunakan untuk menggambarkan interaksi antara sistem yang akan dikembangkan dengan entitas luar.
2. Membuat DFD Level 1

   DFD Level 1 digunakan untuk menggambarkan modul-modul yang adadalam sistem yang akan dikembangkan. DFD Level 1 merupakan hasil *breakdown* DFD Level 0 yang sebelumnya sudah di buat.
3. Membuat DFD Level 2

   Modul-modul pada DFD Level1 dapat di-*breakdown* menjadi DFD Level 2. Modul mana saja yang harus di-*breakdown* lebih detail tergantung pada tingkat kedetailan modul tersebut. Apabila modul tersebut sudah cukup detail dan rinci maka modul tersebut sudah tidak perlu untuk di-*breakdown* lagi. Untuk sebuah sistem, jumlah DFD Level 2 sama dengan jumlah modulpada DFD Level 1 yang di-*breakdown*.
4. Membuat DFD Level 3 dan seterusnya

   DFD Level 3, 4, 5, dan seterusnya merupakan *breakdown* dari modul pada DFD Level di-atasnya. *Breakdown* pada DFD Level 3, 4, 5, dan seterusnya aturannya sama persis dengan DFD Level 1 atau Level 2.

Pada suatu diagram DFDsebaiknya jumlah modul tidak boleh lebih dari 20 buah. Jika lebih dari 20 buah modul, diagram akan terlihat rumit dan susah untuk di baca sehingga menyebabkan sistem yang dikembangkan jugamenjadi rumit.

2.2.12 Kamus Data

Kamus data (data *dictionary*) dipergunakan untuk memperjelas aliran data yang digambarkan pada DFD. Kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukkan (*input*) dan keluaran (*output*) dapat dipahami secara umum (memiliki standar cara penulisan). Kamus data biasanya berisi:

1. nama – nama dari data
2. digunakan pada – merupakan proses-proses yang terkait data
3. deskripsi – merupakan deskripsi data
4. informasi tambahan – seperti tipe data, nilai data, batas nilai data, dan komponen yang membentuk data

Komponen data memiliki beberapa simbol untuk menjelaskan informasi tambahan sebagai berikut:

Simbol	Keterangan
=	di susun atau terdiri dari
+	dan
[|]	baik … atau ...
$\{\}^n$	n kali di ulang/bernilai banyak
( )	data opsional
…	batas komentar

Gambar 2.7 Simbol Kamus Data

Kamus data pada DFD nanti harus dapt dipetakan dengan hasil perancangan basis data yang dilakukan sebelumnya. Jika ada kamus data yang tidak dapat dipetakan pada tabel hasil perancangan basis data berarti hasil perancangan basis data dengan perancangan dengan DFD masih belum sesuai, sehingga harus ada yang di perbaiki baik perancangan basis datanya, perancangan DFD-nya, atau keduanya.

2.2.13 SSL (*Secure Socket Layer)*

Keamanan-keamanan untuk aplkasi *web* mencakup *Secure-HTTP* dan *Secure Socket Layer*, yang menyediakan otentikasi untuk *server-server* dan *browser-browser*, juga meningkatkan tingkat kepercayaan (*confidentially*) dan integritas datauntuk komunikasi yang berlangsung antara *server web* dan *browser*. *S-HTTP* secara spesifik di rancang untuk mendukung protocol HTTP *(Hypertext Transfer Protocol*) dalam hal otorisasi dan keamanan dokumen. SSL menawarkan metode perlindungan yang mirip, tetapi melindungi saluran komunikasi di antara

2 protokol bagian bawah dalam tumpukan protocol, yaitu antara lapisan aplikasi dan lapisan jaringan dan transpor menurut standar protocol TCP/IP. Selain hal di atas, SSL dapat juga digunakan untuk transaksi-transaksi selain yang berjalan di *web*, tetapi tidak di rancang untuk menangani keputusan keamanan berbasis pada otentikasi pada peringkat aplikasi atau dokumen. Ini berarti harus menggunakan metode yang lain untuk mengendalikan akses ke berkas-berkas (*file*) yang berbeda.

2.2.14 Mengenal Web Server

Web server adalah sebuah perangkat lunak server yang berfungsi menerima permintaan *HHTP* atau *HTTPS* dari klien yang di kenal dengan *web browser* dan mengirimkan kembali hasilnya dalam bentu halaman-halaman webyang umumnya berbentuk dokumen *HTML*. *Web server* yang terkenal diantaranya adalah *Apache* dan *Microsoft Internet Information Service* (ISS).

Apache (Server HTTP Apache atau *server Web/WWW/Apache*) adalah *web server* yang dapat dijalankan di banyak system operasi *(Unix, BSD, Linux, Microsoft Windows dan Novell Netware* serta *platform* lainnya) yang berguna untuk melayani dan memfungsikan situs *web*. Protokol yang digunakan untuk melayani fasilitas *web/www* ini menggunakan *HTTP*.

Sedangkan IIS (*Internet Information Service*) adalah sebuah *HTTP web server* yang digunakan dalam sistem operasi *server Windows*, mulai dari *Windows NT 4.0 Server, Windows 2000 Server* atau *Windows Server 2003*. Layanan ini merupakan layanan terintegrasi dalam *Windows 2000 Server, Windows Server 2003* atau sebagai add-on dalam *Windows NT 4.0*. Layanan ini berfungsi sebagai

pendukung protocol TCP/IP yang berjalan dalam lapisan aplikasi (*application layer*). IIS juga menjadi fondasi dari *platform Internet* dan *Internet Microsoft*, yang mencakup *Microsoft Site Server, Mivrosoft Commercial Internet System* dan produk-produk *Microsoft BackOffice* lainnya[7].

2.2.15 *Search Engine Optimalization* (SEO)

Untuk membuat web lebih cepat terdaftar (terindex) *search engine*, wajib melakukan SEO. Berikut adalah beberapa trik-trik SEO yang mendasar:

1. Menggunakan *meta* yang lengkap dalam *tag head* dari sebuah halaman *website*.
2. Setiap halaman menggunakan *meta tag* yang berlainan. Jangan membuat perulangan *keyword* pada *website*.
3. Menggunakan *alt image* (deskripsi gambar) pada setiap gambar.
4. Tidak melakukan *farming link*. Di sini berarti, tidak menampung *link-link* yang sangat banyak dalam sebuah halaman *website*.
5. Tetap rajin untuk meng-*update website*. Meskipun kecil, ini akan sangat berarti dalam pencarian *website* oleh *search engine*.

2.2.16 Macromedia Dreamweaver 8

Salah satu *HTML web authoring* yang popular digunakan oleh para *webmaster* yaitu *Macromedia Dreamweaver 8.* Macromedia Dreamweaver mempunyai tool-tool yang lengkap untuk membuat sebuah *website* secara sempurna. Fitur *editing* secara visual yang dimiliki oleh program ini memungkinkan untuk membuat *web* tanpa harus mengenal kode-kode yang ada didalamnya, sehingga proses pembuatan *web* akan semakin praktis dan cepat.

2.2.17 *World Wide Web* (WWW)

Salah satu layanan aplikasi di Internet ini adalah *World Wide Web* (WWW), pelayanan yang cukup baru dikembangkan di Internet dan menjadi layanan aplikasi yang paling popular digunakan pemakai jaringan Internet dan perkembangannya terus dilakukan sampai saat ini untuk menyempurnakan teknologi ini. WWW atau yang biasa disebut dengan *Web* saja, bekerja menggunakan teknologi yang disebut *hypertext*, yang kemudian dikembangkan menjadi suatu protokol aplikasi yang disebut HTTP (*Hypertext Transfer Protocol*). Dengan adanya fasilitas ini menjadikan *Web* sebagai salah satu aplikasi yang paling luwes untuk menjelajahi *Internet.* Dengan menggunakan WWW, peng-aksesan beragam sumber informasi di *Internet* misalnya *gopher, WAIS, ftp, mail* dan sebagainya, dapat dilakukan melalui suatu cara yang menggabungkan beberapa jenis representasi dan metode pengaksesan informasi dan menyajikannya dalamberagam bentuk informasi seperti text, grafik, suara, animasi, video, dan sebagainya.

2.2.18 *Electronic Mail* (*E-Mail*)

Aplikasi ini adalah yang paling banyak digunakan, dan termasuk salah satu dari aplikasi pertama di Internet. Dengan *E-mail*, dapat mengirim dan menerima surat, pesan, dokumen, secara elektronik dengan pemakai lain di Internet yang mempunyai alamat *e-mail*.

2.2.19 Pengenalan *Website*

Website adalah sebuah media presentasi online untuk sebuah perusahaan atau individu. *Website* juga dapat digunakan sebagai media penyimpanan

informasi secara online, seperti *detik.com, okezone.com, vivenews.com,* dan lain-lain.

Berdasarkan kedinamisan, *website* dapat dibedakan menjadi dua, yaitu *website statis* dan *website dinamis*. *Website statis* bersifat *un-real time.* Maksudnya *website statis* tidak dapat di ubah secara simultan dalam sebuah *web based administrator,* sehingga hanya bisa di ubah ketika langsung mengakses file pada server dan melakukan perubahan pada *source code*nya. Sedangkan *website dinamis* adalah sebuah *website* yang dilengkapi sebuah pengaturan (*administrator*) yang dapat berfungsi sebagai sarana untuk mengubah data *website* secara *real time* tanpa menyentih *source code* dalam *server*.

Untuk menciptakan *website* dinamis ini diperlukan bahasa pemograman yang nantinya akan berfungsi untuk memproses data yang dimasukkan, kemudian akan diteruskan ke *database*, selanjutnya akan ditampilkan pada halaman *web*. Beberapa contoh bahasa pemograman *web* yang dapat digunakan diantaranya adalah *PHP, ASP, JSP* dan *ColdFusion*.

Di sini dalam pembuatan *website* memilih *PHP*, karena *PHP* dapat dikatakan paling mudah dimengerti dibandingkan bahasa-bahasa pemograman *web* lainnya. Meskipun, dalam penggunannya *PHP* sangat menguntungkan, tetapi masih banyak juga kalangan yang menganggap *PHP* sulit, terutama kalangan pemula yang mengiginkan kepraktisan dalam meng-generate kode-kode *PHP*. Namun, sekarang hal ini sudah dapat diwujudkan dengan adanya *tools Recordset* dari *Macromedia Dreamweaver*[7].

2.2.20 Algoritma A Priori

Algoritma a priori adalah jenis asosiasi pada *data mining*. Aturan yang menyatakan asosiasi antara beberapa atribut sering disebut *affinity analysis* atau *market basket analysis*.

Analisis asosiasi atau *assosiation rule mining* adalah teknik *data mini*ng untuk menemukan aturan *asosiatif* antara suatu kombinasi *item*. Contoh aturan *asosiatif* dari analisisi pembelian di suatu pasar swalayan adalah dapat diketahuinya berapa besar kemungkinan seorang pelanggan membeli roti bersamaan dengan susu. Dengan pengetahuan tersebut, pemilik pasar swalaya dapat mengatur penempatan barangnya atau merancang kampanye pemasaran untuk kombinasi barang tertentu.

Penting tidaknya suatu aturan asosiatif dapat diketahui dengan dua parameter, yaitu *support* dan *confidence*. *Support* (nilai penunjang) adalah persentase kombinasi *item* tersebut dalam *database*, sedangkan *confidence* (nilai kepastian) adalah kuatnya hubungan antar-item dalam aturan asosiasi[5].