BAB 2

TINJAUAN PUSTAKA

2.1 Tinjauan Perusahaan

Tinjauan perusahaan adalah sebuah penelitian tentang sejarah, visi dan misi, struktur organisasi, serta deskripsi jabatan yang ada pada perusahaan yang akan menjadi tempat dibangunnya sistem penjualan berbasis *web* ini nantinya.

Perusahaan yang diteliti adalah toko Aurora Shoes, yang bergerak dalam bidang penjualan sepatu. Kegiatan utama yang dilakukan toko Aurora Shoes adalah melakukan penjualan sepatu. Berikut uraian sejarah berdirinya toko Aurora Shoes, visi dan misi, struktur organisasi serta deskripsi jabatan di toko Aurora Shoes.

2. 1.1. Sejarah Perusahaan

Toko Aurora Shoes didrikan pada bulan Februari tahun 2011, toko Aurora Shoes merupakan perusahaan yang bergerak dalam bidang penjualan produk barang, yang akan membantu dalam penyediaan produk untuk mendukung kebutuhan produk barang pada perorangan, baik dalam bentuk produk barang ataupun *item* pembantu lainnya.

Toko Aurora Shoes merupakan perusahaan baru dan sedang berkembang, oleh karena itu tahap selanjutnya yang sangat penting bagi toko Aurora Shoes adalah membangun *image* atau *brand* perusahaan yang positif untuk mendapatkan kepercayaan pelanggan. *Image* yang positif akan sangat membantu toko Aurora Shoes untuk tetap menjadi pilihan utama bagi pelanggan serta tetap bertahan

dalam persaingan bisnis penjualan sepatu. Toko Aurora Shoes mempunyai layanan *retur* yang baik, sehingga rencana dari perorangan dalam mengoptimalkan pengadaan produk barang dan meningkatkan kualitas pelayanan dapat tercapai dengan baik pula.

2. 1.2. Visi dan Misi Perusahaan

a. Visi Aurora Shoes

"Menjadi perusahaan penyedia sepatu yang berkualitas dengan layanan terbaik dalam memberikan solusi terpercaya dan bernilai tambah bagi *customer*.

b. Misi Aurora Shoes

"Mengembangkan perusahaan dengan kultur profesional, integritas dan berorientasi bisnis. Mengembangkan ragam produk dan teknologi yang berdaya guna meningkatkan *benefit* dan *value* bagi *customer*,secara berkelanjutan serta menyusun dan melakukan kontrol pada dengan tetap fokus pada kualitas dan komitmen."

2. 1.3. Logo Aurora Shoes

Toko Aurora Shoes dalam menjalankan bisnis penjualan sepatu selama ini, selalu menggunakan logo atau bendera perusahaan. Berikut ini (Gambar 2.1) adalah logo dari toko Aurora Shoes :



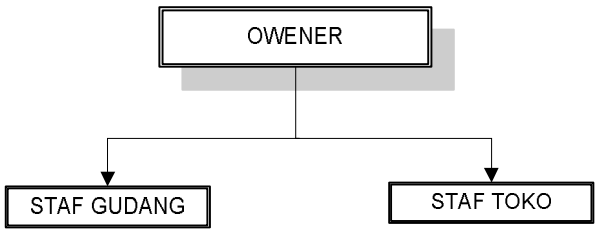
Gambar 2.1 Logo Aurora Shoes

2. 1.4. Struktur Organisasi Aurora Shoes

Struktur organisasi (Gambar 2.2) perusahaan merupakan suatu gambaran yang menunjukkan suatu jabatan atau pekerjaan yang harus dilaksanakan oleh masing- masing bagian atau anggota dari organisasi perusahaan yang tergambar dengan struktur.

Struktur organisasi toko Aurora Shoes dipimpin oleh seorang manejer yang selaku *owner* yang dibantu oleh pegawai toko dan pegawai gudang.

1. *OWNER*
2. STAF TOKO
3. STAF GUDANG



Gambar 2.2 Struktur Organisasi Toko Aurora Shoes

2. 1.5. Deskripsi Jabatan

Berikut deskripsi jabatan, tugas, dan wewenang dari struktur organisasi di toko Aurora Shoes :

1. Owener

Owener merupakan pemilik modal atau saham tertinggi toko Aurora Shoes. Owener dapat disebut juga pemilik perusahaan.

2. Staf Toko

Staf toko merupakan orang yang bekerja di toko Aurora Shoes bertanggung jawab menjaga toko serta melayani *costumer* yang datang langsung ke toko Aurora Shoes, dan mempunyai wewenang untuk mengelola laporan transaksi.

3. Staf Gudang

Staf gudang merupakan orang yang bekerja di toko Aurora Shoes bertanggung jawab menjaga gudang toko Aurora Shoes, dan mempunyai wewenang untuk mengelola laporan stok barang dan laporan pengadaan barang.

2.2 Landasan Teori

2.2.1 Pengenalan E-Commerce

E-Commerce (*electronic commerce*) merupakan satu teknologi yang berkembang pesat seiring dengan kehadiran Internet dalam kehidupan kita. Banyak orang mendapatkan manfaat kemudahan berbisnis melalui Internet.

Electronik Commerce menggambarkan cakupan yang luas mengenai teknologi, proses, dan praktek yang dapat melakukan transaksi bisnis tanpa menggunakan kertas sebagai sarana mekanisme transaksi. Hal ini bisa dilakukan dengan berbagai cara seperti melalui *email*, *Electronic Data Interchange* (EDI), atau bisa juga *Wide World Web*. *Electronic Commerce* ini juga meliputi transaksi di dalam dan diantara sektor bisnis yang khusus (*private*) dan umum (*public*), serta sistem yang melibatkan komunitas dalam negeri maupun internasional.

E-commerce merupakan bagian dari *e-business*. Dimana *e-commerce* mengacu pada Internet untuk belanja *online* dan jangkauannya sempit. Sebagai

sub perangkat dari *e-business*. Cara pembayaran pada *e-commerce* dilakukan melalui transfer uang secara digital melalui *account paypal* atau kartu kredit. Sedangkan e-bisnis mengacu pada Internet tapi jangkauannya lebih luas. Hubungan bisnis terjadi ketika perusahaan atau individu berkomunikasi dengan klien atau nasabah melalui *email* akan tetapi pemasaran atau penjualan dilakukan melalui Internet. Cara ini dapat memberikan keuntungan fleksibilitas dan efisiensi. Pembayaran dilakukan melalui pembayaran digital dengan menggunakan *e-gold* yang sudah diakui di seluruh dunia dalam melakukan transaksi *online*.

2.2.2 Definisi *e-commerce*

Definisi *e-commerce* menurut kamus Wikipedia adalah Perdagangan elektonik atau e-dagang (bahasa Inggris: *electronic commerce*) yaitu penyebaran, pembelian, penjualan, pemasaran barang dan jasa melalui sistem elektronik seperti internet atau televisi, www, dan jaringan komputer lainnya [2]. E-dagang dapat melibatkan transfer dan elektronik, pertukaran data elektronik, sistem manajemen inventaris otomatis, dan sistem pengumpulan data otomatis.

Menurut Dian Andriana, *e-commerce* merupakan suatu cara berbelanja atau berdagang secara *online* atau *direct selling* yang memanfaatkan fasilitas Internet dimana terdapat website yang dapat menyediakan layanan "*get and deliver*".

Definisi *e-commerce* yang sudah distandarkan dan disepakati bersama adalah "*E-commerce is a dynamic set of technologies, applications, and business process that link enterprise, consumers, and communities through electronic transactions and the electronic exchange of goods, services, and information*". *E-*

Commerce merupakan satu set dinamis teknologi, aplikasi dan proses bisnis yang menghubungkan perusahaan, konsumen, dan komunitas tertentu melalui transaksi elektronik dan perdagangan barang, pelayanan, dan informasi yang dilakukan secara elektronik [2]. (David Baum)

Dengan demikian, dapat disimpulkan bahwa *e-commerce* merupakan proses pembelian dan penjualan jasa atau produk antara dua belah pihak melalui Internet (*Commerce net*) dan sejenis mekanisme bisnis elektronik dengan fokus pada transaksi bisnis berbasis individu dengan menggunkan internet sebagai media pertukaran barang atau jasa baik antar instansi atau individu dengan instansi (*Net-Ready*).

2.2.3 Manfaat *e-commerce*

Banyak manfaat dari penjualan secara *online* melalui media Internet, yaitu dengan adanya *e-commerce* suatu organisasi atau perusahaan dapat menjalankan bisnisnya untuk menjangkau seluruh Indonesia bahkan sampai ke luar negeri, beberapa manfaat *e-commerce* antara lain sebagai berikut :

1. Kemampuan grafis Internet mampu memperlihatkan produk apa adanya (natural) serta dapat membuat brosur berwarna dan menyebarkannya tanpa ongkos kirim.
2. Lebih aman membuka toko *online* dibanding membuka toko biasa.
3. Berjalan di dunia maya Internet tidak mengenal hari libur dan hari besar, semua transaksi bisa dilakukan kapan saja dan dimana saja.
4. Tanpa batas-batas wilayah dan waktu, sehingga memberikan jangkauan pemasaran yang luas dan tak terbatas oleh waktu.

5. *Revenue stream* (arus pendapatan) yang baru yang mungkin sulit atau tidak dapat diperoleh melalui cara konvensional.
6. Meningkatkan *market exposure* (pangsa pasar), dimana penggunaan *e-commerce* memungkinkan untuk meningkatkan pangsa pasar yang semula mempunyai pangsa pasar di dalam negeri saja, dengan adanya *e-commerce* maka pangsa pasar menjangkau luar negeri.
7. Menurunkan biaya operasi (*operating cost*). Penggunaan teknologi Internet memungkinkan kita untuk melakukan kegiatan perdagangan selama 24 jam sehari, 7 hari seminggu, akan tetapi tidak terpengaruh terhadap biaya yang dikeluarkan untuk biaya lembur karyawan atau pegawai, karena segala sesuatunya dikerjakan oleh komputer yang tidak membutuhkan operator untuk menjalankan proses perdagangan, cukup hanya dengan penggunaan *software* tertentu maka semua aktivitas dalam transaksi perdagangan dapat dilakukan.
8. Penghematan besar yang dimungkinkan melalui *e-mail*. Penghematan ini terjadi karena berkurangnya penggunaan kertas dalam segala proses transaksi, dimana segala sesuatunya didalan *e-commerce* menggunakan data digital sehingga tidak membutuhkan kertas sebagai media yang pada akhirnya memberikan penghematan besar terhadap pengeluaran dalam proses transaksi.

2.2.4 Internet dan *e-commerce*

Dalam beberapa waktu terakhir ini, dengan begitu merebaknya media Intenet dimana-mana. Khususnya di Indonesia, dengan semakin menjamurnya warnet yang bertebaran hampir seperti wartel, ternyata telah membuat banyak

perusahaan mulai mencoba menawarkan berbagai macam produk mereka dengan menggunakan media ini.

Kehadiran Internet yang walaupun masih merupakan industri baru yang dalam fase pertumbuhahan, yang masih terus berubah seta penuh ketidakpastian, telah memperkokoh keyakinana akan pentingnya teknologi dalam pencapaian tujuan finansial perusahaan melalui modifikasi dan efisiensi proses bisnis, yaitu dengan memanfaatkan *E-Commerce*. Kemampuan Internet untuk menjangkau pelanggan baru dan penghematan biaya yang cukup signifikan untuk distribusi dan pelayanan pelanggan merupakan keuntungan yang bisa didapat perusahaan dengan memindahkan roda *commerce* ke media Internet.

Perkembangan teknologi Internet yang sangat cocok berubah menjadi strategi atau model bisnis yang cocok hari ini mungkin tidak relevan untuk diterapkan besok. Didalam lingkungan Internet yang kompetitif ini, di samping perencanaan yang baik serta harapan yang realistis terhadaop teknologi baru, juga diperlukan kesiapan infrastruktur yang cukup memadai dan visi yang komperhensif untuk bisa berjalan melewati ketidakpastian yang begitu cepat.

E-Commerce merupakan salah satu keunggulan dari Internet, hingga akhirnya di era sekarang ini (banyak disebut era digital) nampaknya tiada hari yang terlewatkan tanpa mendengar atau membaca kata *E-Commerce* di berbagai media informasi.

2.2.5 Sistem transaksi *e-commerce*

Untuk saat ini sistem transaksi di *e-commerce* bisa dibilang sudah lengkap dan terbilang aman untuk melakukan transaksi. Karena dengan adanya keamanan

ini pelanggan dapat tertarik untuk melakukan transaksi secara *online* di Internet. Beberapa model pembayaran *online* di Internet untuk saat ini yang masih sering digunakan sebagai berikut :

1. Paypal

Paypal merupakan salah satu alat pembayaran (*payment process*) teraman dan terbanyak digunakan di dunia Internet. Saat ini, orang Indonesia sudah dapat memindahkan uang yang berada di *account Paypal* langsung ke rekeningnya. Pengguna Internet dapat membeli barang di *ebay,* lisensi *software original*, keanggotaan situs, urusan bisnis, mengirim dan menerima donasi atau sumbangan dan mengirimkan uang ke pengguna *Paypal* lain di seluruh dunia. *Paypal* mengatasi kekurangan dalam pengiriman uang tradisional seperti cek atau *money order* yang prosesnya memakan waktu yang lama. *Paypal* tidak berbeda dengan rekening bank. Hal ini pertama yang perlu Anda lakukan adalah membuat *account* tersebut dengan dana dari kartu kredit atau mentransfer dana dari *account Paypal* orang lain ke *balance Paypal* Anda. Jika proses ini sudah diselesaikan, maka *account Paypal* sudah dapat digunakan untuk bertransaksi.

Pada bulan Februari 2008 *Paypal* sudah menerima 190 negara dan 16 mata uang. Pengguna kartu *Paypal* di Indonesia masih harus menggunakan hitungan US dollar karena rupiah masih belum ada di *Paypal*. *Ebay* adalah tempat jual-beli *online* terbesar di dunia. Anda dapat membeli dan mencari apapun, mulai dari barang elektronik, kendaraan hingga barang-barang langka dari seluruh pengguna *ebay* di dunia. Cara pembayaran utama di *ebay* adalah

dengan *Paypal* karena lebih aman dan otomatis, meskipun penjual yang lain juga menerima pembayaran dengan cara lain.

2. Kartu Kredit

Kartu kredit dapat digunakan sebagai alat pembayaran dengan terlebih dahulu mendaftarkan diri Anda sebagai konsumen pada *web* tersebut. Selain kartu kredit sebagai alat pembayaran alternatif, Anda bisa menggunaka *e-cash,* yaitu sebuah *account* khusus untuk pembayaran melalui Internet. *Account* tersebut dibuka dengan kartu kredit yang dimiliki sebelumnya. Konsumen hanya perlu mengisi *account e-cash*-nya untuk digunakan.

3. Smartcard

Di Singapura, *smartcard* dikenal dengan istilah *cash card*. Pemakaian *smartcard* hampir sama dengan pemakaian kartu ATM yang biasa digunakan untuk berbelanja. Pada saat transaksi, uangnya di-*debet* langsung dari *account* di bank. Untuk pembayaran di Internet, konsumen harus memiliki *smart card reader*. Dalam penggunaanya, alat khusus ini disambungkan ke *port serial* di komputer. Ketika melakukan transaksi, kartu *smartcard* harus digesekan pada alat tersebut agar *chip* yang terdapat di kartu tersebut dapat dibaca oleh komputer. Penggunaan sistem pembayaran ini menggunakan *software* bernama *e-wallet*. Contoh *website* yang telah menerima *smartcard* sebagai alat pembayaran adalah http://www.discvault.com.

4. iCheck

iCheck adalah metode pembayaran menggunakan cek. Pembayaran ini membutuhkan nomor cek milik konsumen.

5. E-Gold

E-gold dapat diperoleh dengan membuka rekening gratis di www.E-gold.com. *E-gold* adalah pembayaran *virtual* yang cukup digemari oleh penjual *e-book* dan *software* di Internet. Dari segi keamanan, ada robot yang selalu mengganti PIN Anda dan mengirimnya ke *e-mail* saat Anda membutuhkannya. *E-gold* dapat dibeli atau dijual pada *web-web* penyedia jenis pembayaran ini, misalnya www.greatachiever.com.

2.3 Internet dan *World Wide Web*

E-Commerce bergantung pada keberadaan Internet sebagai media utama bertransaksi. Tanpa Internet, maka perdagangan secara elektronik ini tidak akan pernah terlaksana.

2.3.1 Pengertian Internet

Interconnected Network atau lebih dikenal sebagai internet lahir sebagai hasil langsung dari keperluan-keperluan penelitian dan pengembangan oleh pemerintah, universitas dan perusahaan besar. Dengan semakin bertambahnya waktu, internet semakin berkembang sehingga para pemakai non-peneliti dapat menikmati keuntungan dari internet. Tujuan utama merealisasikan internet tersebut adalah untuk dapat bekrja bersama-sama secara efisien, sehingga memungkinkan untuk *sharing* data dan informasi secara bersama-sama.

Setiap komputer dan jaringan secara langsung maupun tidak langsung terhubung ke beberapa jalur utama yang disebut internet *backbone*. Setiap komputer dan jaringan yang terhubung tersebut dibedakan satu dengan yang lainnya menggunakan nama yang unik (*unique name*) yang disebut alamat IP (IP

Address) 32 bit. Komputer dan jaringan tersebut bisa memiliki *platform* yang berbeda, missal Unix, Linux, Windows dan Macc. Hal ini memerlukan adanya standar aturan/protokol yang harus disepakati agar setiap komputer dan jaringan tersebut dapat saling berkomunikasi. *Protocol* standar yang digunakan di internet adalah TCP/IP (*Transmission Control Protocol/Internet Protocol*).

Terdapat beberapa alasan mengapa internet semakin popular saat ini, yaitu:

1. Informasi dapat di-*share* dengan pertukaran pesan antar komputer di seluruh dunia
2. Banyaknya orang yang membutuhkan informasi tentang suatu hal, namun secara geografis sangat berjauhan untuk memperoleh informasi tersebut
3. Kualitas informasi sendiri adalah sangat kompleks, teknis dan selalu *up-to-date*.
4. Waktu sebagai faktor sangat penting, baik dari nilai informasi tersebut atau ketersediaan informasi yang dibutuhkan.

2.3.2 Definisi Internet

Internet merupakan jaringan global komputer dunia, besar dan sangat luas sekali dimana setiap komputer saling terhubung satu sama lainnya dari negara ke negara lainnya di seluruh dunia dan berisi berbagai macam informasi, mulai dari text, gambar, audio, video, dan lainnya.

Internet itu sendiri berasal dari kata *Interconnection Networking*, yang berarti hubungan dari banyak jaringan komputer dengan berbagai tipe dan jenis, dengan menggunakan tipe komunikasi seperti telepon, salelit, dan lainnya.

2.3.3 Sejarah Internet

Internet pertama kali dikembangkan oleh ARPANet, suatu proyek yang dimulai dari Departemen Pertahanan Amerika Serikat (*US Departement of Defense–DOD*) pada tahun 1969, suatu percobaan dalam *reliable networking* (jaringan secara terpercaya) untuk menghubungkan antara DOD dengan kontraktor penelitian militer, termasuk sejumlah besar universitas yang melakukan penelitian dengan dana militer. Pada konsepnya sebenarnya internet merupakan suatu jaringan, yang mana suatu paket informasi dapat dikirim dari suatu komputer ke komputer yang lain. Hal ini dimaksudkan untuk mempermudah pemindahan berbagai data penting apabila terjadi perang

ARPA merupakan singkatan dari *Advanced Research Projects Agency*, cabang dari *Defence* (Pertahanan) yang mempunyai kewajiban membagi-bagikan uang *grant* sehingga menjadi DARPA (*Defence-ARPA*). ARPANet mulai dengan 3 komputer kecil yang dikoneksi di California digabungkan dengan satu di Utah, tetapi secara cepat berkembang di seluruh kontinen. Internet ini mulai tumbuh pesat pada dekade 1990.

2.3.4 Layanan dan fasilitas internet

Ada beberapa hal yang dapat dilakukan di internet, seiring dengan perkembangan teknologi internet dan infrastrukturnya maka fasilitas dan layanan semakin terus berkembang dan bertambah. Berikut ini beberapa layanan yang tersedia di internet:

1. *Electronic Mail* (E-mail)

   E-*mail* adalah surat atau pesan elektronik yang dikirimkan dan diterima oleh dan antar individu atau komputer.

2. *Mailing List*

   Mailing List yaitu layanan internet sebagai pengembangan dari *email* yang difungsikan untuk berdiskusi.

3. *Inter Relay Chat* (IRC)

   Inter Relay Chat adalah fasilitas yang diguna kan untuk melakukan perbincangan atau bercakap-cakap melalui internet menggunakan teks atau sering disebut dengan *chatting*.

4. *File Tranfer Protocol* **(**FTP)

   File Transfer Protocol adalah layanan *Internet* untuk melakukan transfer file antara komputer *Client* dengan *Server* di *Internet*.

5. *Tele Networking* (TelNet)

   Telnet adalah program untuk mengakses dan menjalankan komputer (*host/server*) dari jauh atau *Remote* login.

6. World Wide *Web*

   WWW adalah layanan internet yang paling banyak dikenal orang dan paling cepat perkembangan teknologinya. Layanan ini menggunakan link *hypertext* yang disebut *hyperlink* untuk merujuk dan mengambil halaman-halaman *web* dari *server*.

7. *Teleconference*

Teleconference adalah fasilitas internet yang juga digunakan untuk berbincang-bincang dengan cara yang kompleks yaitu mulai dari suara hingga gambar.

2.3.5 Pengertian *World Wide Web*

World Wide Web (WWW) merupakan salah satu layanan yang didapat oleh pemakai komputer yang terhubung ke Internet. *Web* pada awalnya adalah ruang informasi dalam Internet, dengan menggunakan teknologi hyperteks, pemakai dituntun untuk menemukan informasi dengan mengikuti link yang disediakan dalam dokumen web yang ditampilkan dalam *browser web.*

World Wide Web sering disingkat WWW atau *web* saja, yaitu sebuah sistem dimana informasi dapat kita nikmati dalam bentuk teks, gambar, suara, dan lain-lain serta ikut dapat diakses oleh perangkat lunak yang disebut *browser*. Informasi di *web* pada umumnya ditulis dalam bentuk HTML. Informasi lainnya disajikan dalam bentuk grafis (dalam format GIF, JPG, PNG), suara (dalam bentuk format AU, WAV), dan objek multimedia lainnya (seperti MIDI, Shockwave, Quicktime Movie, 3D World). WWW dijalankan dalam *server* yang disebut HTTPD.

2.3.6 *Universal Resource Locator* (URL)

URL adalah konsep nama file standar yang diperluas dengan jaringannya. Nama file ini tidak hanya menunjukan direktori dan nama filenya saja, tetapi juga nama mesinnya dalam jaringan. URL dapat disediakan (ada atau diakses) dengan berbagai metode dan bisa jadi bukan sekedar file. Karena URL dapat menunjukan

queri-queri dokumen dapat disimpan dalam database, hasil dari suatu perintah finger atau achie atau apapun yang berkaitan dengan data hasil proses.

2.3.7 *Domain Name Service* (DNS)

DNS digunakan untuk mempermudah penggunaan internet, dengan memetakan IP *address* ke nama host. Agar data nama host dapat didistribusikan di banyak server, format data yang digunakan harus mencerminkan terdistribusinya data tersebut. Untuk itu, digunakan format tree dengan masing-masing nodenya disebut domain. Penulisan nama host secara lengkap disebut sebagai *Full Qualifield Domain Name (FQDN).*

DNS bekerja secara hirarki dan berbentuk seperti pohon (tree). Bagian atas adalah *Top Level Domain* (TLD) seperti COM, ORG, EDU, MIL dsb. Seperti pohon DNS mempunyai cabang-cabang yang dicari dari pangkal sampai ke ujung. Pada waktu kita mencari alamat misalnya linux.or.id pertama-tama DNS bertanya pada TLD server tentang DNS Server yang melayani domain .id misalnya dijawab ns1.id, setelah itu dia bertanya pada ns1.id tentang DNS Server yang bertanggung jawab atas .or.id misalnya ns.or.id kemudian dia bertanya pada ns.or.id tentang linux.or.id dan dijawab 64.29.24.175

Sedangkan untuk mengubah IP menjadi nama *host* melibatkan domain in-addr.arpa. Seperti domain lainnya domain in-addr.arpa pun bercabang-cabang. Yang penting diingat adalah alamat IP-nya ditulis dalam urutan terbalik di bawah in-addr.arpa. Misalnya untuk alamat IP 64.29.24.275 prosesnya seperti contoh linux.or.id: cari server untuk arpa, cari server untuk in-addr.arpa, cari server untuk 64.in-addr.arpa, cari server 29.64.in-addr.arpa, cari server untuk 24.29.64.in-

addr.arpa. Dan cari informasi untuk 275.24.29.64.in-addr.arpa. Pembalikan urutan angkanya memang bisa membingungkan.

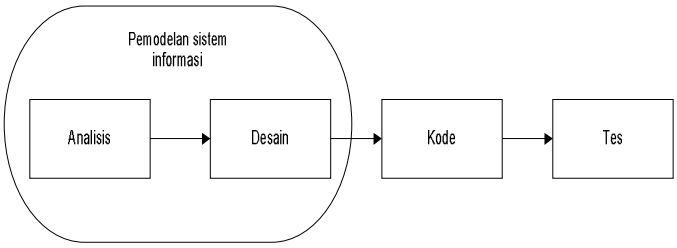
2.3.8 HTTP (*Hypertext Transfer Protocol*)

Server dan *browser web* berkomunikasi satu sama lain dengan protokol yang memang dibuat khusus untuk ini yaitu HTTP. HTTP bertugas menangani permintaan-permintaan *(request)* dan *browser* untuk mengambil dokumen-dokumen *web*. HTTP bisa dianggap sebagai sistem yang bermodel *client-server*.

2.4 SDLC *Model Sequential Linear*

System Development Life Cycle (SDLC) adalah suatu metodologi yang digunakan untuk mengembangkan, memelihara dan menggunakan perangkat Lunak (O'briend, 2001), metodologi penelitian ini mencakup sejumlah fase atau tahapan.

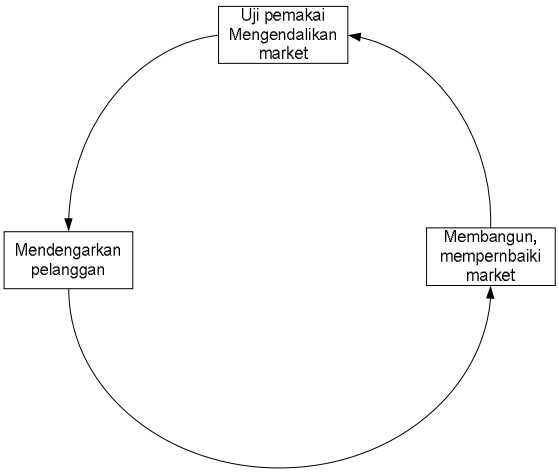
Linear Sequential Model (model sekuensial linear) dapat dilihat pada gambar 2.1, ditemukan pertama kali oleh Royce. Model ini sering disebut model klasik atau *waterfall.* Model ini menyarankan pendekatan pengembangan secara sekuen dan sistematik untuk pengembangan perangkat lunak [3]. Model ini merupakan model tertua. Model ini terdiri atas beberapa tahapan, yaitu rekayasa, dan pemodelan sistem informasi, analisis kebutuhan perangkat lunak, desain, generasi kode, pengujian dan pemeliharaan.



Gambar 2.3 Model Sequensial Linear

Berikut metode pengembangan perangkat lunak lainnya :

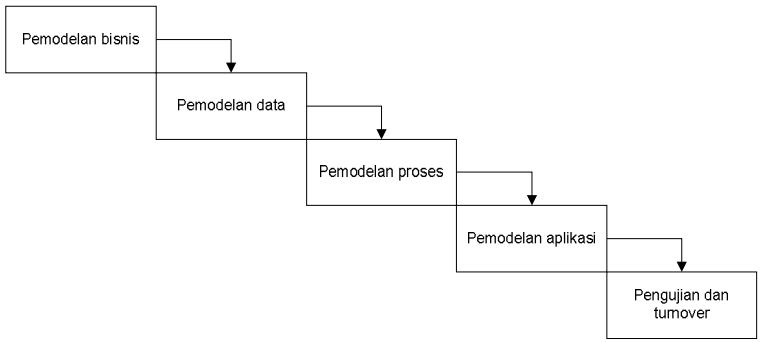
1. *Prototyping Model* (model paradigma) dimulai dengan pengumpulan kebutuhan, pengembangan dan pelanggan bertemu dan mendefinisikan objek keseluruhan dari perangkat lunak, mengidentifikasi segala kebutuhan yang diketahui dan area garis besar dimana definisi lebih jauh merupakan keharusan, kemudian dilakukan perancangan kilat [3]. Perancangan kilat membawa kepada konstruksi sebuah prototipe yang kemudian dievaluasi oleh pemakai. Dipakai untuk menyaring kebutuhan pengembangan perangkat lunak. Siklus dari model prototipe dapat dilihat pada gambar 2.4 dibawah ini :



Gambar 2.4 Model Prototipe

2. *Rapid Application Development* (RAD), adalah sebuah model proses pengembangan perangkat lunak sekuensial linear yang menekankan siklus pengembangan yang sangat pendek. Model RAD merupakan adaptasi

"berkecepatan tinggi" dari linear sequential model pengembangan yang cepat dapat diperoleh dengan menggunakan pendekatan konstruksi berbasis komponen [3]. Lihat gambar 2.5 berikut :



Gambar 2.5 Model *Rapid Application Development* (RAD)

3. *Evolutionary Software Process Model* adalah perulangan. Model ini dicirikan denga pengembang versi-versi sistem yang lebih lengkap sedikit demi sedikit [3]. Model telah mempertimbangkan untuk mengkoordinasikan evaluasi produk, secara lengkap model ini terdiri dari :
   a. *Incremental model*, model ini mengkombinasikan antara linear sequential model dengan filosofi iteratif pada *prototyping*. Pada masing-masing sekuen linear, yang menghasilkan perangkat lunak yang semakin meningkat kompleksitasnya.
   b. *Sipral model*, model ini digunakan oleh Boehm. Model ini menggabungkan antar sifat alami iterasi dati *prototyping* dengan aspek sistematik dan terkendali dari linear sequential model. Model ini memberi peluang untuk pengembangan cepat.

2.5 Teori Basis Data

2.5.1 Definisi basis data

Basis data terdiri dari 2 kata, yaitu basis dan data, basis dapat diartikan sebagai maskas atau gudang tempat bersarang atau berkumpul [4]. Sedangkan data adalah representasi fakta dunia nyata yang mewakili suatu objek seperti manusia (pegawai, siswa, pembeli dan lain-lain), barang hewan, peristiwa, konsep keadaan dan sebagainya yang direkam dalam bentuk angka, huruf, simbol, teks, gambar, bunyi, atau kombinasinya. Basis data sendiri dapat didefinisikan dalam sejumlah sudut pandang, seperti :

1. Himpunan kelompok data (*arsip*) yang saling berhubungan yang diorganisasikan sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.
2. Kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (redudansi) yang tidak perlu untuk memenuhi berbagai kebutuhan.
3. Kumpulan file, tabel, atau arsip yang saling berhubungan yang akan disimpan dalam media penyimpanan elektronis.

Basis data dalam lemari arsip sesungguhnya memiliki prinsip kerja dan tujuan yang sama. Prinsip utamanya adalah pengeluaran data dan tujuan utamanya adalah kemudahan dan kecepatan dalam pengembalian data. Perbedaannya hanya terletak pada media penyimpanan yang digunakan. Maka basis data menggunakan media penyimpanan elektronik seperti disk (disket atau *hardisk*). Hal ini karena lemari arsip langsung dikelola oleh manusia sementara basis data dikelola melalui perantara alat atau mesin pintar elektronis (yang kita kenal komputer).

Satu hal yang harus diperhatikan, bahwa basis data bukan hanya sekedar tidak menyimpan data secara elektronis (dengan bantuan komputer), artinya tidak semua bentuk penyimpanan data secara elektronis bisa disebut basis data. Kita dapat menyimpan dokumen berisi data dalam file teks (dengan program pengolah data), *file spread* sheet dan lain-lain. Tetapi tidak bisa disebut basis data karena didalamnya tidak ada pemilahan dan pengelompokan data sesuai jenis dan fungsi data. Yang sangat ditonjolkan basis data adalah pengaturan, pengelompokan, pengorganisasian data yang akan kita simpan sesuai dengan fungsinya.

2.5.2 Operasi dasar basis data

Didalam sebuah disk, basis data dapat diciptakan dan dapat pula ditiadakan. Sementara dalam sebuah basis data, kita dapat menempatkan satu atau lebih file ataupun table. Setiap basis data pada umumnya dibuat untuk mewakili sebuah basis data sebuah semesta data yang spesifik.

Operasi-operasi yang dapat kita lakukan berkenaan dengan basis data dapat meliputi :

1. Pembuatan basis data baru (*create database*), yang identik dengan pembuatan lemari arsip yang baru.
2. Penghapusan basis data (*drop database*), yang identik dengan perusakan lemari arsip (sekaligus dengan isinya, jika ada).
3. Pembuatan file atau tabel dari suatu basis data (*create table*), yang identik dengan penambahan map arsip baru ke sebuah lemari arsip yang telah ada.
4. Penghapusan file atau tabel dari suatu basis data (*drop table*), yang identik dengan perusakan map arsip lama yang ada di sebuah lemari arsip.

5. Penambahan atau pengisian data baru kesebuah file atau tabel disebuah basis data (*insert*) yang identik dengan penambahan lembaran arsip kesebuah map arsip.
6. Pengambilan data dari sebuah file atau tabel (*retrieve atau search*), yang identik dengan pencarian lembaran arsip yang ada disebuah map arsip.
7. Pengubahan data dari sebuah file atau tabel (*update*).

2.5.3 *Database Management System (DBMS)*

Berikut ini akan dikemukakan definisi dari *Database Management System* (*DBMS*) :

"Pengelolaan basis data secara fisik tidak dilakukan oleh pemakai secara langsung, tetapi ditangani oleh sebuah perangkat lunak (sistem) yang khusus atau spesifik yang menerapkan mekanisme pengamanan data, pemakaian data secara bersama, pemaksaan keakuratan atau konsistensi data, dan sebagainya".

"DBMS dapat diartikan sebagai perangkat lunak yang menentukan bagaimana data diorganisasi, disimpan, diubah, dan diambil".

DBMS dapat menjadi alternatif penggunaan secara khusus untuk aplikasi, semisal penyimpanan data dalam file dan menulis kode aplikasi yang spesifik untuk pengaturannya.

Penggunaan *DBMS* yang bertujuan untuk menyediakan fasilitas antarmuka (interface) dalam melihat atau menikmati data (*user oriented*) kepada pemakai yang tergantung pada kemampuan dan dukungan *DBMS* yang beroperasi secara efisien. Sehingga agar bisa menggunakan *DBMS* dengan baik, perlu diketahui cara kerja dari *DBMS* tersebut. Pendekatan yang dilakukan untuk menggunakan *DBMS*

secara baik meliputi : implementasi *DBMS* dan arsitektur secara mendetail untuk dapat memahami desain dari suatu basis data

2.6 Pemodelan Data

Pemodelan sistem memainkan peranan yang penting dalam pengembangan sistem. Pemodelan data kadang-kadang disebut pemodelan *database* karena model data kadang-kadang diimplementasikan sebagai sebuah *database*. Pemodelan data dapat digambarkan dengan ERD (*Entity Relationship Diagram*).

2.6.1 *Entity Relationship Diagram (ERD)*

Entity Relationship Diagram adalah suatu model atau teknik pendekatan yang dapat menyatakan atau menggambarkan hubungan *entity* didalam sebuah sistem, dimana hubungan tersebut dinyatakan sebagai *one-to-one*, *one-to-many* dan *many-to-many*. [5]

Entity relationship bertujuan untuk menentukan objek-objek data (*Entity*) dan *relationship* (hubungan) yang ada pada objek-objek tersebut.

Komponen *Entity Relationship Diagram* :

1. *Entity*

Entity merupakan individu yang mewakili sesuatu yang nyata (eksistensinya) dan dapat dibedakan dari sesuatu yang lain.

2. Atribut

Atribut adalah yang mendeskripsikan karakteristik (properti) dari entitas tersebut. Penentuan atau pemilihan atribut-atribut yang relevan bagi sebuah entitas merupakan hal penting lainnya dalam pembentukan model data.

3. Relationship

Relationship adalah menunjukkan hubungan diantara sejumlah entitas yang berasal dari himpunan entitas yang berbeda. Simbol yang digambarakan untuk menggambarkan *relationship* pada teknik *entity relationship diagram* adalah berupa belah ketupat.

Setiap *relationship* mempunyai derajat *relationship*, yaitu tingkat atau hubungan yang terdiri dari tiga derajat, yaitu :

a. *Unary* (derajat 1)

Suatu *relationship* yang dihubungkan dengan suatu *entity* set, dimana penghubungnya ada dua.

b. *Binary* (derajat 2)

Relationship derajat dua adalah *relationship* yang biasa terjadi yaitu dua *entity* set dihubungkan dengan satu *entity relationship*.

c. *Ternary* (derajat 3)

Relationship derajat tiga.

Relationship diagram juga mempunyai *Cardinalitas* yaitu batasan pada jumlah *entity* yang terdiri dari:

1. Relasi satu ke satu (*one-to-one*)

Adalah setiap entitas pada himpunan entitas A berhubungan dengan paling banyak satu entitas pada himpunan yang lain.

2. Relasi satu ke banyak (*one-to-many*)

Adalah yang berarti setiap entitas pada himpunan A dapat berhubungan dengan banyak entitas pada himpunan entitas lain, begitu juga sebaliknya dimana

setiap entitas pada himpunan entitas lainnya berhubungan dengan paling banyak dengan satu entitas pada himpunan yang satu.

Adalah suatu relasi yang terjadi dengan batasan relasi yang berupa banyak berbanding banyak.

2.7 Analisis Sistem

Penggambaran dan analisis model sistem yang digunakan untuk menggambarkan pembangunan perangkat lunak adalah analisis terstruktur. Analisis terstruktur adalah aktivitas pembangunan model dengan menggunakan notasi yang sesuai dengan prinsip analisis operasional. Analisis terstruktur bukan merupakan metode tunggal yang diaplikasikan secara konsisten oleh semua yang menggunakannya. Pada model ini analisis terhadap perangkat lunak dapat digambarkan dalam bentuk *Flow Map*, Diagram konteks, *Data Flow Diagram* (DFD) dan Kamus Data.

2.7.1 *Flowmap*

Flowmap adalah representasi grafik dari sistem informasi, proses-proses, aliran-aliran data logis, masukan, keluaran, dan *file-file* serta entitas-entitas sistem operasi yang berhubungan dengan sistem informasi [5].

2.7.2 Diagram konteks

Diagram Konteks adalah bagian dari DFD yang berfungsi memetakan model model lingkungan, yang dipresentasikan dengan lingkaran tunggal yang

mewakili keseluruhan sistem. Diagram ini adalah diagram *level* tertinggi dari DFD. Diagram konteks menyoroti sejumlah karateristik 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 kedunia luar.
4. Penyimpanan Data (*storage*), yaitu digunakan secara bersama antara sistem dengan terminator. Data ini dapat dibuat oleh sistem dan digunakan oleh lingkungan atau sebaliknya dibuat oleh lingkungan dan digunakan oleh sistem. Hal ini berarti pembuatan simbol *data storage* dalam diagram konteks dibenarkan, dengan syarat simbol tersebut merupakan bagian dari luar sistem.
5. Batasan, yaitu batasan antara sistem dan lingkungan.

2.7.3 *Data Flow Diagram (DFD)*

DFD atau singkatan dari *Data Flow Diagram* merupakan representasi grafik dari suatu sistem yang menunjukan proses atau fungsi, aliran data, tempat penyimpan data dan *entitas eksternal*. DFD juga digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan. Dengan menggunakan DFD, rancangan yang akan kita buat akan lebih terarah dan lebih rinci. Sehingga kita tidak akan mengalami kesulitan dalam melakukan perancangan. *Data flow diagram* memiliki empat komponen, antara lain akan dijelaskan dibawah ini [6] :

1. *Terminator (external exitity)*

   Terminator mewakili *entity external* yang berkomunikasi dengan sistem yang sedang dikembangkan. *Terminator* merupakan kesatuan dilingkungan sistem. Biasanya terminator ini dikenal dengan nama entitas (*external*) sumber atau tujuan (*source* dan *sink*). Terminator dapat juga berupa departemen, divisi atau sistem diluar sistem yang berkomunikasi dengan sistem yang dikembangkan.

2. Proses

   Proses sering dikenal dengan nama *Bubble*, fungsi atau informasi. Komponen proses menggambarkan bagian dari sistem yang mentransformasikan *input* ke *output*, atau dapat dikatakan bahwa komponen proses menggambarkan transformasi satu *input* atau lebih menjadi *output*. Setiap proses harus diberikan penjelasan yang lengkap sebagai berikut :

   a. Identifikasi Proses

      Umumnya berupa angka yang menunjukan nomor dari proses atau ditulis pada bagian atas simbol proses.

   b. Nama Proses

      Menunjukan apa yang dikerjakan oleh proses tersebut. Nama proses harus jelas dan lengkap menggambarkan bagian prosesnya nama proses diletakan dibawah identifikasi proses.

1. Penyimpanan data *(data store)*

   Data store digunakan sebagai sarana untuk mengumpulkan data. *Data store* disimbolkan dengan dua garis horizontal yang paralel dimana tertutup pada salah satu ujungnya atau dua garis horizontal saja. *Data store* ini

biasanya berkaitan dengan penyimpanan-pemyimpanan seperti *file* atau *database* yang berkaitan dengan penyimpanan secara komputerisasi, contohnya *file* pita magnetic, *file* disket atau *file harddisk*. *Data store* juga berkaitan dengan pemyimpanan data.

2. Alur Data *(Data Flow)*

Suatu data *flow* atau alur data dapat dipresentasikan dengan anak panah yang menunjukan arah menuju ke dan keluar dari suatu proses. Alur data ini digunakan untuk menerangkan perpindahan data atau satu paket data atau informasi dari suatu bagian sistem ke bagian lainnya. Selain menunjukan arah, alur data pada model yang dibuat dapat merepresentasikan bit, karakter, pesan, formulir, bilangan *real* dan macam-macam informasi yang berkaitan dengan komputer.

2.7.4 Kamus data

Kamus data adalah suatu daftar atau kamus dari seluruh elemen-elemen data yang diperlukan oleh suatu sistem. Kamus data dibuat berdasarkan arus data yang terdapat pada DFD. Arus data dalam DFD bersifat global sehingga hanya dapat ditunjukkan nama arus datanya saja dan keterangan lebih lanjut mengenai struktur arus data dapat dilihat dari kamus data [5].

2.8 Normalisasi

Normalisasi adalah suatu teknik dalam menstruktur data dalam cara-cara tertentu untuk mengurangai atau mencegah timbulnya masalah yang berhubungan dengan pengolahan data dalam *database*. Normalisasi juga diartikan sebagai

proses pengelompokan data elemen menjadi tabel-tabel yang menunjukan entitas dan relasinya.

Konsep-konsep pada normalisasi, antara lain :

1. Kunci atribut (*Key field* atau *Key attribut*) yautu suatu kunci *field* yang mewakili *record* atau *tupple*
2. Kunci kandidat (*Candidate key*) yaitu suatu atribut atau satu set atribut yang mengidentifikasi secara unik suatu *entity.*
3. Kunci primer (*Primary key*) yaitu satu atribut atau satu set atribut yang yang mengidentifikasi secara unik dan mewakili setiap kejadian pada satu *entity*.
4. Kunci alternatif (*Alternate key*) yaitu kunci kandidat yang dipakai sebagai kunci primer.
5. Kunci tamu (*Foreign key*) yaitu suatu atribut atau satu set atribut dan melengkapi hubungan yang menunjukan ke induknya.

Bentuk-bentuk normalisasi yaitu :

1. Normal satu (1NF atau *First Normal Form*)

   Relasi berada pada normal kesatu jika semua atribut mempunyai nilai yang bersifat atomic.

2. Normal kedua (2NF atau *Second Normal Form*)

   Relasi berada pada normal kedua jika relasi tersebut merupakan normal satu dan atribut bukan merupakan kunci tergantung penuh pada kunci primer.

3. Normal ketiga (3NF atau *Three Normal Form*)

Relasi berada pada normal ketiga jika relasi tersebut merupakan normal kedua dan atribut bukan kunci tidak tergantung secara transitif pada kunci primer.

4. BCNF (*Boyce Cood Normal Form*)

   Relasi berada pada BCNF jika dan hanya jika faktor penentunya adalah kunci kandidat dan relasi tersebut merupakan normal ketiga.

2.9 *Hypertext 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 Microsof Frontpage, Adobe GoLive, Macromedia Dreamweaver, Ultradev dan lain-lainnya. Namun untuk seorang web develover kemampuan dasar perintah HTML tetap dibutuhkan.

Homepage dibangun oleh sebuah file yang didalamnya terdapat kode-kode dan informasi dan dibaca oleh sebuah aplikasi yang bernama browser. Oleh browser, kode dan informasi akan disajikan sebagai sebuah halaman web yang kita kenal sebagai homepage. Kode tersebut haruslah ditulis menurut suatu aturan tertentu yang dapat dimengerti browser. Aturan tersebut dikenal dengan "bahasa pengkodean".

Untuk membuat homepage website, bahasa pengkodean yang digunakan adalah HTML. Sekalipun banyak orang menyebutnya sebagai salah satau bahasa pemrograman, HTML sebenarnya sama sekali bukan bahasa pemrograman karena HTML merupakan bahasa Mark Up (pengkodean) yang digunakan untuk menentukan format atau style dan teks yang ditandai. Simbol Mark Up yang digunakan oleh HTML ditandai dengan tanda lebih kecil (<) dan lebih besar (>), keduanya dinamai tag. Sebagai contoh akan ditampilkan teks yang tercetak tebal, maka mark up yang digunakan adalah sebagai berikut:

<B> Teks ini bercetak tebal </B>

Tag <HTML> harus diletakan pada bagian awal dan tag </HTML> harus diletakan pada bagian terakhir dari suatu file HTML. Tag-tag HTML tidak bersifat case sensive artinya penggunaan huruf besar maupun huruf kecil tidak menjadi masalah. Jadi tag <HTML> akan sama dengan tag <html>.

2.10 *Hypertext Preprocessor* (PHP)

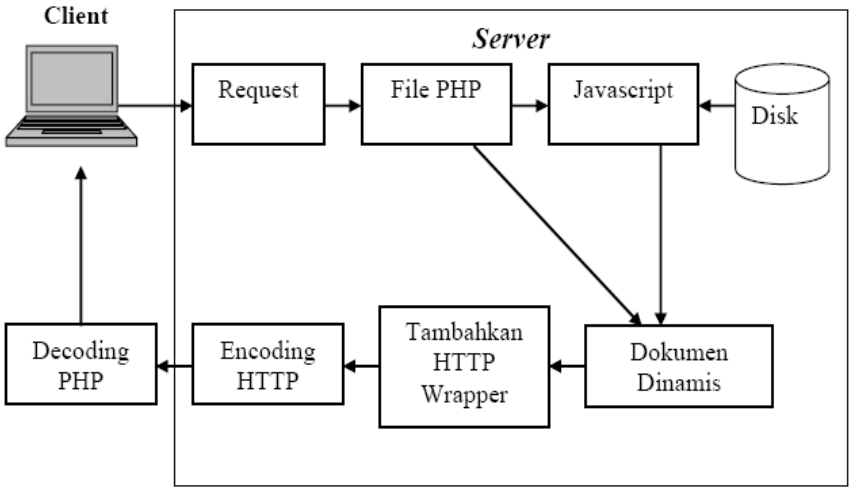
PHP (Personal Home Page Tools) adalah sebuah bahasa *scripting* yang dibundel dengan HTML, yang berjalan disisi *server*. Sebagian besar perintahnya berasal dari bahasa *C, Java* dan *Perl* dengan beberapa tambahan fungsi khusus PHP.

Bahasa ini memungkinkan para pembuat aplikasi web yang menyajikan HTML yang dinamis dan interaktif dengan cepat dan mudah, yang di hasilkan server. PHP biasa berinteraksi dengan hampir semua teknologi *web* yang sudah ada. *Developer* biasa menulis sebuah program PHP yang mengeksekusi suatu program CGI di *server* lain. Fleksibilitas ini amat bermanfaat bagi pemilik situs-

situs web yang besar dan sibuk, karena pemilik masih menggunakan aplikasi-aplikasi yang sudah terlanjur dibuat dimasa lalu dengan menggunakan CGI, ISAP atau dengan *script* seperti *Perl, awk* atau *Phyton* selama proses migrasi ke aplikasi baru yang di buat dengan menggunakan PHP.

Kode program PHP menyatu dengan tag-tag HTML dalam satu file. Kode PHP diawali dengan tag <? Dan ditutup dengan tag ?>. File yang berisi tag HTML dan kode PHP ini diberi ekstensi .php atau ekstensi lainnya yang ditetapkan pada *Apache/web server*. Berdasarkan ektensi ini, pada saat file diakses, server akan tahu bahwa file ini mengandung kode PHP. Server akan menerjemahkan kode ini dan menghasilkan output dalam bentuk tag HTML yang akan dikirim ke *browser client* yang mengakses file tersebut.

Dengan PHP, *developer* tidak perlu lagi berurusan dengan dua buah file yang terpisah seperti pada CGI. *Browser web* mengacu secara langsung ke file yang dituju, yang lalu dibaca oleh *server* sebagaimana file HTML statis biasa. Bedanya, sebelum dikirim balik ke *browser web, server web* memeriksa isi file dan menentukan apakah ada kode didalam file tersebut yang harus dieksekusi. Bila ada, kode-kode tersebut akan dieksekusi. Hasilnya akan dimasukkan kedalam dokumen yang sama. *Server web* bekerja secara langsung terhadap file yang bersangkutan tidak memanggil *script* terpisah seperti pada metode CGI. Seluruh kode di eksekusi di *server* (oleh karena itu disebut *server-side script*). Lihat gambar 2.6 dibawah ini untuk melihat kerja dari PHP dan Apache.



Gambar 2.6 Prinsip Kerja PHP dan *Apache*

PHP membuat proses pengembangan aplikasi menjadi mudah karena kelebihan-kelebihannya, yaitu :

1. *Script* (kode program) terintegrasi dengan file HTML, sehingga *developer* biasa berkonsentrasi langsung pada penampilan webnya.
2. Tidak ada proses *compiling* dan *linking.*
3. Berorientasi objek *(object Oriented).*
4. Sintaksis pemrogramannya udah dipelajari, sangat menyerupai *C* dan *Perl.*
5. Integrasi yang sangat luas ke berbagai *server database*. Menulis *web* yang terhubung ke *database* menjadi sangat sederhana. Database yang didukung oleh PHP : *Oralce, Sybase, mSQL, MySQL, Solid, ODBC, PostgreSQL, Adabas D, FilePRo, Velocis, Informic, dBase, UNIX dbm.*

2.11 *Cascading Style Sheets* (CSS)

Cascading Style Sheets (CSS) adalah suatu bahasa *stylesheet* yang digunakan untuk mengatur tampilan suatu dokumen yang ditulis dalam bahasa *markup*. Penggunaan yang paling umum dari CSS adalah untuk memformat

halaman *web* yang ditulis dengan HTML dan XHTML. Walaupun demikian, bahasanya sendiri dapat dipergunakan untuk semua jenis dokumen XML termasuk SVG dan XUL. Spesifikasi CSS diatur oleh World Wide Web Consortium (W3C).

CSS digunakan oleh penulis maupun pembaca halaman web untuk menentukan warna, jenis huruf, tata letak, dan berbagai aspek tampilan dokumen. CSS digunakan terutama untuk memisahkan antara isi dokumen (yang ditulis dengan HTML atau bahasa markup lainnya) dengan presentasi dokumen (yang ditulis dengan CSS). Pemisahan ini dapat meningkatkan aksesibilitas isi, memberikan lebih banyak keleluasaan dan kontrol terhadap tampilan, dan mengurangi kompleksitas serta pengulangan pada stuktur isi.

CSS memungkinkan halaman yang sama untuk ditampilkan dengan cara yang berbeda untuk metode presentasi yang berbeda, seperti melalui layar, cetak, suara (sewaktu dibacakan oleh browser basis-suara atau pembaca layar), dan juga alat pembaca *braille*. Halaman HTML atau XML yang sama juga dapat ditampilkan secara berbeda, baik dari segi gaya tampilan atau skema warna dengan menggunakan CSS.

2.12 JavaScript

JavaScript adalah bahasa *script* yang digunakan pada halaman HTML. *JavaScript* adalah varian dari *java* yang sangat terkenal dalam lingkungan pemrograman *web*. *JavaScript* mempunyai keunggulan pada sisi *client*, maupun diakses lebih cepat dari *script server*. Tetapi kelemahannya pada *script* ini belum tentu mampu diproses oleh *browser client* tergantung kompetibilitas *browser* tersebut. Cara untuk menjalankan j*avascript* hanya dengan *javascript-enabled*

yaitu *browser* yang mampu menjalankan *javascript* seperti *Netscape Navigator* (versi 2.0 keatas) atau *internet ekplorer* (MSIE-versi 3.0 ke atas).

Beberapa hal mengenai JavaScript :

1) *JavaScript* didesain untuk menambah interaktif suatu *web*.
2) *JavaScript* merupakan sebuah bahasa *scripting* dengan bahasa pemrograman yang ringan.
3) *JavaScript* biasanya disisipkan (*embedded*) dalam halaman HTML.
4) *JavaScript* adalah bahasa interpreter (yang berarti skrip dieksekusi tanpa proses kompilasi.

Ada beberapa cara untuk menuliskan JavaScript yaitu :

a) *Script* pada bagian *head*

Skrip ini dieksekusi ketika dipanggil (biasanya berbentuk *function*) atau dipanggil berdasarkan *event* tertentu. Peletakan *script* di *head* akan menjamin *script* di-*load* terlebih dahulu sebelum digunakan.

```
<html>
<head>
<script type=”text/javascript”>
Function message ()
{
Alert (“Alert ini dipanggil saat even dijalankan”);
}
<script>
</head>
</html>
```

b) *Script* pada *Body*

Skrip akan dieksekusi ketika halaman di-*load* sampai dibagian <body> dan ketika ditempatkan berarti antara isi dan *javacript* dijadikan satu bagian.

```
<html>
<head>
</head>
<body>
<script type=”text/javascript”>
Document.write(“Pesan ini akan tampil”);
</script>
</body>
</html>
```

c) External JavaScript

Untuk memudahkan menulis *script* yang berulang-ulang di setiap halaman maka *JavaScript* dapat ditulis di *file* secara *ekternal*. Jadi, antara HTML dan *JavaScript* dipisah, kemudian *file* tersebut dipanggil dari dokumen HTML. *File JavaScript* tersebut disimpan dengan ektensi .js.

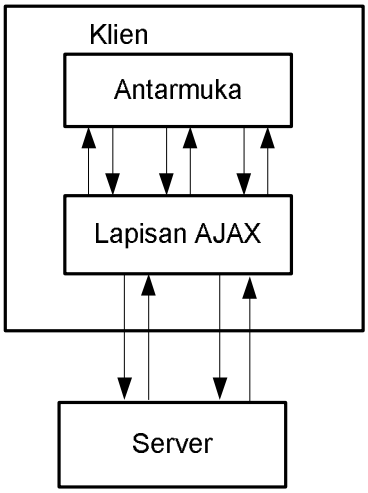
```
<html>
<head>
</head>
<body>
<script src=”xxx.js”>
</script>
```

```
</body>
</html>
```

2.13 *Asynchronous JavaScript and XML* (AJAX)

AJAX (*Asynchronous JavaScript and XML*) sesungguhnya bukanlah suatu bahasa pemrogarman, melainkan hanya suatu teknik yang memungkinkan untuk membuat aplikasi *web* yang interaktif. Dengan menggunakan AJAX, aplikasi *web* dapat berinteraksi dengan server di latar belakang sehingga tidak mempengaruhi halaman *web* secara keseluruhan. Sebagai gambaran, Anda mungkin sering mendaftarkan diri pada suatu situs *web* sebagai anggota baru. Setelah Anda selesai mengetikan *user ID* saat registrasi, sistem bisa memberitahukan bahwa nama yang Anda pakai sudah digunakan oleh orang lain atau belum.

Bagaimana aplikasi *web* ini bisa tahu hal itu padahal Anda belum mengklik tombol apapun dalam halaman *web*. Rahasianya terletak pada pemrosesan di latar belakang yang dilakukan oleh AJAX. Gambar 2.7 Menunjukan aplikasi *web* yang berbasis AJAX.



Gambar 2.7 Model Aplikasi Web dengan AJAX

Nampak bahwa pada aplikasi *web* yang berbasis AJAX terdapat bagian yang disebut lapisan AJAX (atau terkadang disebut mesin AJAX). Lapisan inilah yang mengatur komunikasi antara *client* dan *server*.

Butir penting yang perlu diperhatikan dalam lapisan AJAX :

1. Lapisan AJAX meminta layanan ke *server* atau tidak, tergantung pada jenis layanan antarmuka. Sebagai contoh, untuk validasi terhadap objek formulir tidak perlu melakukan permintaan ke *server*. Cukup ditangani di sisi *client*. Namun untuk pemeriksaan keabsahan *password*, mau tidak mau harus dilayangkan ke *server* karena harus berhubungan dengan database.
2. Sekalipun penambahan lapisan AJAX berkesan memperlambat proses (karena komunikasi ke server tidak lagi dilakukan secara langsung), pada kenyataanya yang terjadi adalah sebaliknya. Hal ini karena permintaan yang dilakukan oleh lapisan AJAX menghasilkan informasi yang jauh lebih sedikit dibanding kalau yang diminta adalah keseluruhan halaman.
3. Lapisan AJAX bekerja dengan pendekatan asinkron. Asinkron berarti bahwa *client* bisa meminta layanan dari *server* dan tidak perlu menunggu sampai *server* melayaninya. Dengan demikian, saat informasi dari *server* belum didapat, layanan kepada pemakai tetap bisa dilakukan.
4. Komunikasi antara *client* dan *server* tidak perlu mengubah keseluruhan antarmuka dalam halaman *web*, tetapi cukup hanya bagian tertentu yang terpengaruh. Hal seperti ini tidak bisa dilakukan pada aplikasi *web* yang tidak menggunakan AJAX.

AJAX pertama kali ditemukakan pada tahun 2005 oleh Jesse James Garret, presiden dan pendiri perusahaan Adaptive Path. Menurut beliau, nama AJAX

diberikan untuk memudahkan dalam berkomunikasi dengan klien daripada kalau menyebutnya "*Asynchronous JavaScript + CSS + DOM + XMLHttp Request*". Teknik ini sebenarnya sudah ada sejak 1999. Pada masa itu Microsoft membuat objek bernama *XMLHttpRequest* (Objek yang kelak akan dilibatkan dalam membuat AJAX) sebagai control *ActiveX* pada Internet Explorer 5. AJAX menjadi sangat popular baru setelah digunakan intensif oleh Yahoo! Dan Google.

Menurut Jesse James Garret, di dalam artikelnya yang berjudul "*AJAX A New Approach to Web Applications*", AJAX yang lengkap mencakup [9]:

1. XHTML (*Extensible HyperText Markup Language*), sebagai peluasan dari HTML.
2. CSS (*Cascading Style Sheets*) sebagai pemformat bagian dari dokumen.
3. DOM (*Document Object Model*) untuk mengatur interaksi dan penampilan isi HTML dan XHTML secara dinamis.
4. XML (*Extensible Markup Language*) memungkinkan untuk melakukan pertukaran data.
5. XSLT (*Extensible Stylesheet Language Transformation*) adalah bahasa berbasis XML yang ditujukan untuk melakukan transformasi dokumen XML ke bentuk yang lain.
6. *XMLHttpRequest* untuk melakukan pertukaran data secara asinkron.
7. *JavaScript* sebagai bahasa untuk membentuk lapisan AJAX.

Memang nampaknya sangat kompleks karena melibatkan banyak hal. Namun tidak pelu khawatir bahwa hal itu akan menyusahkan dalam mewujudkannya. Dalam praktik, AJAX bisa diimplementasikan tanpa menggunakan XML, XSLT, maupun CSS.

Software yang digunakan untuk membangun aplikasi *web* dengan AJAX :

1. *Browser* (misalnya *Internet Explorer* dan *Mozilla Firefox*) pada sisi klien.
2. *Web Server* (misalnya *Apache*) pada sisi server.

Tentu saja yang dicantumkan di atas adalah kebutuhan minimal untuk membuat aplikasi *web* yang melibatkan AJAX. Jika aplikasi juga melibatkan database, di sisi server juga perlu dipasang *database server*. Selain itu, agar *web server* dapat berkomunikasi dengan *database server*, perantara seperti PHP diperlukan.

2.14 MySQL

MySQL adalah multi *user database* yang menggunakan bahasa *Strured Query Language* (SQL). MySQL merupakan database server multi user dan multi *threaded* yang tangguh. Dengan memiliki banyak feature MySQL bisa bersaing dengan datgabase komsersial sekalipun. Tidak mengejutkan MySQL menjadi database pilihan untuk banyak pengguna PHP. MySQL termasuk RDBMS (*Relational Database Management System*) itulah sebabnya istilah seperti tabel, baris dan kolom digunakan pada MySQL. Pada MySQL sebuah database mengandung satu atau sejumlah tabel. Tabel terdiri dari sejumlah baris dan setiap baris mengandung satu atau beberapa kolom.

MySQL mampu menangani data yagn cukup besar. Perusahaan yang mengembangkan MySQL yaitu TcX, mengaku menyimpan data lebih dari 40 *database*, 10.000 tabel dan sekitar 7 juta baris, totalnya kurang lebih 100 *Gigabyte* data. SQL adalah sebuah standar yang digunakan untuk mengakses *database*

server. Bahasa ini pada awalnya dikembangkan oleh IBM, namun telah diadopsi dan digunakan sebagai standar industry.

Dengan menggunakan MySQL, proses akses *database* menjadi lebih *user-friendly* dibandingkan dengan menggunakan *dBASE* atau *Clipper* yang masih menggunakan perintah-perintah pemrograman. MySQL merupakan *software database* yang masih yang masih popular di lingkungan Linux, kepopuleran ini sangat ditunjang oleh performa *query* dari databasenya yang saat ini bisa dikatakan paling cepat dan jarang bermaslah. MySQL ini juga sudah berjalan pada lingkungan windows.