

BAB III

LANDASAN TEORI

3.1 Intranet

Menurut Prakoso (2007:119), Intranet adalah sebuah kumpulan jaringan komputer lokal yang menggunakan perangkat lunak internet dan protokol TCP/IP atau HTTP. Oleh karena itu, sebuah jaringan intranet memiliki semua fasilitas yang dimiliki oleh internet seperti *e-mail*, *File Transfer Protocol (FTP)*, dan lain sebagainya. Jaringan intranet merupakan jaringan internet yang hanya dimiliki oleh perusahaan dan tidak dapat diakses dari luar.

Sebuah intranet terdiri atas banyak jaringan komputer lokal yang saling terhubung satu dengan yang lainnya. Kemudian, intranet pun meliputi koneksi melalui satu atau lebih *gateway* komputer menuju dunia internet. Tujuan utama sebuah intranet adalah membagi informasi perusahaan dan sumber daya komputer. Intranet pun digunakan sebuah perusahaan untuk mengelola isi, mengarahkan, menerbitkan, dan mengarsip dokumen. Lebih lanjut, intranet dapat digunakan untuk mendukung kerja kelompok dan *teleconference*.

3.2 Web

Sebuah situs web (sering pula disingkat menjadi situs saja: web site, site) adalah sebutan bagi sekelompok halaman web (web page), yang umumnya merupakan bagian dari suatu domain (domain name) atau subdomain di World Wide Web (WWW) di internet. WWW terdiri dari seluruh situs web yang tersedia kepada publik. Halaman-halaman sebuah situs web diakses dari sebuah URL yang

menjadi “akar” (root), yang disebut homepage (halaman induk; sering diterjemahkan menjadi “beranda”, “halaman muka”), dan biasanya disimpan dalam server yang sama. Tidak semua situs web dapat diakses dengan gratis.

Beberapa situs web memerlukan pembayaran agar dapat menjadi pelanggan, misalnya situs-situs yang menampilkan pornografi, situs-situs berita, layanan surat elektronik (email), dll.

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

Halaman dari website bisa diakses melalui sebuah URL yang biasa disebut *homepage*. URL ini mengatur halaman-halaman situs untuk menjadi sebuah hirarki, meskipun, *hyperlink* yang ada di halaman tersebut mengatur para pembaca dan memberitahu mereka susunan keseluruhan dan bagaimana arus informasi berjalan.

Beberapa website membutuhkan subskripsi (data masukan) agar para user bisa mengakses sebagian atau keseluruhan isi website tersebut. Contohnya, ada beberapa situs-situs email gratisan, yang membutuhkan subskripsi agar kita bisa mengakses situs tersebut.

Penemu website adalah Sir Timothy John “Tim” Berners-Lee, sedangkan website yang tersambung dengan jaringan, pertamakali muncul pada tahun 1991. Maksud dari Tim ketika membuat website adalah untuk mempermudah tukar menukar dan memperbaharui informasi kepada sesama peneliti di tempat dia berkerja. Pada tanggal 30 April 1993, CERN (tempat dimana Tim bekerja) menginformasikan bahwa WWW dapat digunakan secara gratis oleh semua orang.

Sebuah website bisa berupa hasil kerja dari perorangan atau individu, atau menunjukkan kepemilikan dari sebuah organisasi, perusahaan, dan biasanya website itu menunjukkan beberapa topik khusus, atau kepentingan tertentu. Sebuah website bisa berisi hyperink yang menghubungkan ke website lain, jadi, kadangkala perbedaan antar website yang dibuat oleh individu perseorangan dengan website yang dibuat oleh organisasi bisnis bisa saja tidak kentara.

Website ditulis, atau secara dinamik dikonversi menjadi HTML dan diakses melalui sebuah program software yang biasa disebut dengan web browser, yang dikenal juga dengan HTTP Client. Halaman web dapat dilihat atau diakses melalui jaringan komputer dan internet, perangkatnya bisa saja berupa Personal Computer, Laptop Computer ataupun Cellphone.

3.3 Desain

Desain biasa di terjemahkan sebagai seni terapan,arsitektur dan berbagai pencapaian kreatif lainnya. Dalam sebuah kalimat, kata “desain” bisa digunakan baik sebagai kata benda atau kata kerja. Sebagai kata benda “desain” digunakan untuk menyebut hasil akhir dari sebuah proses kreatif, baik itu berwujud sebuah rencana, proposal, atau berbentuk obyek nyata.

3.4 Company Profile Interaktif

Company Profile dapat diartikan sebagai identitas perusahaan. Jika dikemas menjadi interaktif, maka nama tersebut berubah menjadi *Company Profile* interaktif. Dengan adanya *Company Profile* interaktif maka perusahaan tersebut dapat dengan mudah memvisualkan identitas perusahaannya dalam bentuk interaktif yang berupa halaman yang saling berhubungan (link) dengan adanya penjelasan-penjelasan tambahan.

3.5 PHP (Hypertext Preprocessor)

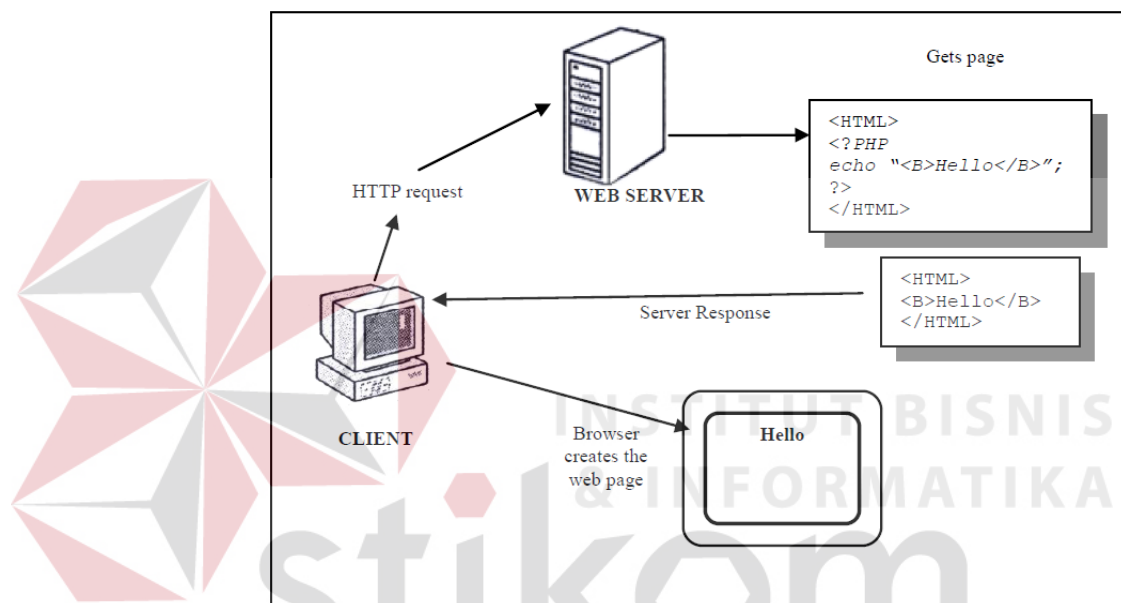
3.5.1 Pengenalan PHP

PHP (akronim dari PHP Hypertext Preprocessor) yang merupakan bahasa pemrograman berbasis web yang memiliki kemampuan untuk memproses data dinamis. PHP dikatakan sebagai sebuah server-side embedded script language artinya sintaks-sintaks dan perintah yang kita berikan akan sepenuhnya dijalankan oleh server tetapi disertakan pada halaman HTML biasa. Aplikasi-aplikasi yang dibangun oleh PHP pada umumnya akan memberikan hasil pada web browser, tetapi prosesnya secara keseluruhan dijalankan di server.

Pada prinsipnya server akan bekerja apabila ada permintaan dari client. Dalam hal ini client menggunakan kode-kode PHP untuk mengirimkan permintaan ke server (dapat dilihat pada gambar dibawah). Ketika menggunakan PHP sebagai server-side embedded script language maka server akan melakukan hal-hal sebagai berikut :

- Membaca permintaan dari client/browser

- Mencari halaman/page di server
- Melakukan instruksi yang diberikan oleh PHP untuk melakukan modifikasi pada halaman/page.
- Mengirim kembali halaman tersebut kepada client melalui internet atau intranet.



Gambar 1.1

Mengapa PHP?

- PHP dapat dijalankan pada platform yang berbeda-beda (Windows, Linux, Unix, etc.)
- PHP merupakan web scripting open source
- PHP mudah dipelajari

3.5.2 Syntax PHP

Kode PHP disimpan sebagai plain text dalam format ASCII, sehingga kode PHP dapat ditulis hampir di semua editor text seperti windows notepad, windows wordpad, dll. Kode PHP adalah kode yang disertakan di sebuah halaman HTML dan kode tersebut dijalankan oleh server sebelum dikirim ke browser.

Pada file .html, HTTP server hanya melewati content dari file menuju ke browser. Server tidak mencoba untuk mengerti atau memproses file, karena itu adalah tugas sebuah browser.

Pada file dengan ekstensi .php akan ditangani secara berbeda. Yang memiliki kode PHP akan diperiksa. Web server akan memulai bekerja apabila berada diluar lingkungan kode HTML. Oleh karena itu server akan melewati semua content yang berisi kode HTML, CSS, JavaScript, simple text di browser tanpa diinterpretasikan di server.

Blok scripting PHP selalu diawali dengan `<?php` dan diakhiri dengan `?>`. Blok scripting PHP dapat ditempatkan dimana saja di dalam dokumen. Pada beberapa server yang mendukung, blok scripting PHP dapat diawali dengan `<?` dan diakhiri dengan `?>`. Namun, untuk kompatibilitas maksimum, sebaiknya menggunakan bentuk yang standar (`<?php ?>`).

Setiap baris kode PHP harus diakhiri dengan semikolon (;). Semikolon ini merupakan separator yang digunakan untuk membedakan satu instruksi dengan instruksi lainnya.

PHP menggunakan // untuk membuat komentar baris tunggal atau /* dan */ untuk membuat suatu blok komentar.

3.5.3 Variabel PHP

Variabel digunakan untuk menyimpan suatu nilai, seperti text, angka atau array. Ketika sebuah variabel dibuat, variabel tersebut dapat dipakai berulang-ulang.

Pada PHP semua variabel harus dimulai dengan karakter '\$'. Variabel PHP tidak perlu dideklarasikan dan ditetapkan jenis datanya sebelum kita menggunakan variabel tersebut. Hal itu berarti pula bahwa tipe data dari variabel dapat berubah sesuai dengan perubahan konteks yang dilakukan oleh user. Secara tipikal, variabel PHP cukup diinisialisasikan dengan memberikan nilai kepada variabel tersebut.

Identifier dalam PHP adalah case-sensitive, sehingga \$text dengan \$Text merupakan variabel yang berbeda. Built-in function dan structure tidak case-sensitive, sehingga echo dengan ECHO akan mengerjakan perintah yang sama. Identifier dapat berupa sejumlah huruf, digit/angka, underscore, atau tanda dollar tetapi identifier tidak dapat dimulai dengan digit/angka. Aturan Penamaan Variabel

- Nama variabel harus diawali dengan sebuah huruf atau garis bawah (underscore) “_”
- Nama variabel hanya boleh mengandung karakter alpha-numeric dan underscore (a-Z, 0-9, dan _)
- Nama variabel tidak boleh mengandung spasi.

3.5.4. String di PHP

Variabel string digunakan untuk nilai yang mengandung karakter string. Hanya ada satu operator string di PHP. Concatenation operator (.). digunakan untuk menggabungkan dua string.

3.5.5. Operator di PHP

Operator digunakan untuk menentukan sebuah nilai dengan melakukan suatu prosedur, atau dengan suatu operasi dari beberapa nilai. Nilai yang digunakan dalam sebuah operasi disebut operand. Penjumlahan adalah sebuah contoh sederhana dari suatu operasi.

3.5.5. 1. Arithmetic Operator (Operator Aritmatika)

PHP menggunakan operator arithmetic dasar :

Operator	Aksi	Contoh	Penejelasan
+	Penjumlahan	$5 + 9$	Menghasilkan nilai : 14
-	Pengurangan	$9 - 4$	Menghasilkan nilai : 5
*	Perkalian	$5 * 6$	Menghasilkan nilai : 30
/	Pembagian	$20 / 4$	Menghasilkan nilai : 5
%	Modulus	$9 \% 4$	Menghasilkan nilai : 1

PHP mengabaikan karakter spasi dalam sebuah operasi. Ekspresi $x = 5 + 9$ dengan dengan $x=5+9$ akan menghasilkan nilai yang sama. Penggunaan spasi disesuaikan dengan keinginan dari para user.

3.5.5. 2. Unary Operator

Tanda minus (-) digunakan dengan sebuah nilai numerik tunggal untuk menegatifkan sebuah bilangan (untuk membuat negatif bilangan positif atau membuat positif bilangan negatif).

3.5.5. 3. Variable Assignment Operator

Tanda sama dengan (=) digunakan untuk mengatur atau menetapkan nilai suatu variabel. Oleh karena itu tanda tersebut dikenal sebagai operator penugasan.

3.5.5.4. Comparison Operator (Operator Perbandingan)

Operator perbandingan digunakan untuk menguji suatu kondisi. Ekspresi yang menggunakan operator perbandingan akan selalu menghasilkan nilai boolean, yaitu antara true (benar) atau false (salah).

Beberapa operator perbandingan yang lain adalah sebagai berikut :

Operator	Arti	Contoh	Menghasilkan benar (true) ketika :
==	Sama dengan	$\$i == \j	$\$i$ dan $\$j$ mempunyai nilai yang sama
<	Kurang dari	$\$i < \j	$\$i$ kurang dari $\$j$
>	Lebih dari	$\$i > \j	$\$i$ lebih dari $\$j$
<=	Kurang dari atau sama dengan	$\$i <= \j	$\$i$ kurang dari atau sama dengan $\$j$
>=	Lebih dari atau sama dengan	$\$i >= \j	$\$i$ lebih dari atau sama dengan $\$j$
!=	Tidak sama dengan	$\$i != \j	$\$i$ tidak sama dengan $\$j$
<>	Tidak sama	$\$i <> \j	$\$i$ tidak sama dengan $\$j$

	dengan		
====	Identik	\$a ==== \$b	Benar jika \$a sama dengan \$b, dan keduanya memiliki type data yang sama.(hanya dalam PHP4)

3.5.5.5. Logical Operator (Operator Logika)

Operator logika digunakan untuk mengkombinasikan kondisi, sehingga beberapa kondisi dapat dievaluasi atau diperiksa dalam sebuah ekspresi. Sebagai contoh logika AND akan bernilai true jika semua kondisi benar. Tabel berikut ini menunjukkan semua anggota dari operator logika.

Operator	Contoh	Bernilai benar jika :
AND / and	\$i && \$j atau \$i AND \$j	\$i dan \$j bernilai benar
OR / or	\$i \$j atau \$i OR \$j	Salah satu atau kedua variabel bernilai benar
XOR	\$i XOR \$j	Salah satu variabel bernilai benar, tetapi tidak keduanya benar
NOT	!\$i	\$i tidak bernilai benar

3.5.5.6. String Concatenation Operator

Tanda titik (.) sebagai operator concatenate (penggabung) digunakan untuk menggabungkan dua atau lebih nilai string menjadi sebuah string tunggal.

3.5.5.7. Variable Assignment Shortcut

Pada PHP dimungkinkan untuk melakukan penggunaan jalan pintas untuk operator pada pernyataan penugasan dimana operand pertama adalah sebuah variabel dan hasilnya disimpan pada variabel yang sama.

Contoh	Ekivalen dengan
<code>\$x += \$y</code>	<code>\$x = \$x + \$y</code>
<code>\$x -= \$y</code>	<code>\$x = \$x - \$y</code>
<code>\$x *= \$y</code>	<code>\$x = \$x * \$y</code>
<code>\$x /= \$y</code>	<code>\$x = \$x / \$y</code>
<code>\$x %= \$y</code>	<code>\$x = \$x % \$y</code>
<code>\$x &= \$y</code>	<code>\$x = \$x & \$y</code>
<code>\$x = \$y</code>	<code>\$x = \$x \$y</code>
<code>\$x ^= \$y</code>	<code>\$x = \$x ^ \$y</code>
<code>\$x .= \$y</code>	<code>\$x = \$x . \$y</code>
<code>\$x >>= 2</code>	<code>\$x = \$x >> 2</code>
<code>\$x <<= 2</code>	<code>\$x = \$x << 2</code>
<code>\$x++</code>	<code>\$x = \$x + 1</code>
<code>\$x--</code>	<code>\$x = \$x - 1</code>

3.6 Database

Menurut Marlinda (2004:1), *database* adalah suatu susunan/kumpulan data operasional lengkap dari suatu organisasi/perusahaan yang diorganisir/dikelola dan disimpan secara terintegrasi dengan menggunakan

metode tertentu menggunakan komputer sehingga mampu menyediakan informasi optimal yang diperlukan pemakainya. Penyusunan satu *database* digunakan untuk mengatasi masalah-masalah pada penyusunan data yaitu redundansi dan inkonsistensi data, kesulitan pengaksesan data, *isolasi* data untuk standarisasi, *multiple user* (banyak pemakai), masalah keamanan (*security*), masalah integrasi (kesatuan), dan masalah data *independence* (kebebasan data).

3.7 Sistem Basis Data

Menurut Marlinda (2004:1), sistem basis data adalah suatu sistem menyusun dan mengelola data-data menggunakan komputer untuk menyimpan atau merekam serta memelihara dan operasional lengkap sebuah organisasi atau perusahaan sehingga mampu menyediakan informasi optimal yang diperlukan pemakai untuk proses mengambil keputusan.

Pada sebuah sistem basis data terdapat komponen-komponen utama yaitu Perangkat Keras (*Hardware*), Sistem Operasi (*Operating System*), Basis Data (*Database*), Sistem Pengelola Basis Data (DBMS), Pemakai (*User*), dan Aplikasi lain (bersifat opsional).

Keuntungan sistem basis data adalah:

1. Mengurangi kerangkapan data, yaitu data yang sama disimpan dalam berkas data yang berbeda-beda sehingga *update* dilakukan berulang-ulang.
2. Mencegah ketidakkonsistenan.
3. Keamanan data dapat terjaga, yaitu data dapat dilindungi dari pemakai yang tidak berwenang.
4. Integritas dapat dipertahankan.

5. Data dapat dipergunakan bersama-sama.
6. Menyediakan *recovery*.
7. Memudahkan penerapan standarisasi.
8. Data bersifat mandiri (*data independence*).
9. Keterpaduan data terjaga, memelihara keterpaduan data berarti data harus akurat. Hal ini sangat erat hubungannya dengan pengontrolan kerangkapan data dan pemeliharaan keselarasan data.

Kerugian sistem basis data adalah:

1. Diperlukan tempat penyimpanan yang besar.
2. Diperlukan tenaga yang terampil dalam mengolah data.
3. Perangkat lunaknya mahal.

3.8 Database Management Sistem

Menurut Marlinda (2004:6), *Database Management System (DBMS)* merupakan kumpulan file yang saling berkaitan dan program untuk pengelolanya. Basis Data adalah kumpulan datanya, sedang program pengelolanya berdiri sendiri dalam suatu paket program yang komersial untuk membaca data, menghapus data, dan melaporkan data dalam basis data.

Bahasa-bahasa yang terdapat dalam DBMS adalah:

1. *Data Definition Language (DDL)*

Pola skema basis data dispesifikasikan dengan satu set definisi yang diekspresikan dengan satu bahasa khusus yang disebut DDL. Hasil kompilasi perintah DDL adalah satu set tabel yang disimpan di dalam file khusus yang disebut *data dictionary/directory*.

2. *Data Manipulation Language (DML)*

Bahasa yang memperbolehkan pemakai mengakses atau memanipulasi data sebagai yang diorganisasikan sebelumnya model data yang tepat.

3. *Query*

Pernyataan yang diajukan untuk mengambil informasi. Merupakan bagian DML yang digunakan untuk pengambilan informasi.

DBMS memiliki fungsi sebagai berikut:

1. *Data Definition*

DBMS harus dapat mengolah pendefinisian data.

2. *Data Manipulation*

DBMS harus dapat menangani permintaan-permintaan dari pemakai untuk mengakses data.

3. *Data Security dan Integrity*

DBMS dapat memeriksa *security* dan *integrity* data yang didefinisikan oleh DBA.

4. *Data Recovery dan Concurrency*

a. DBMS harus dapat menangani kegagalan-kegagalan pengaksesan basis data yang dapat disebabkan oleh kesalahan sistem, kerusakan disk, dan sebagainya.

b. DBMS harus dapat mengontrol pengaksesan data yang konkuren yaitu bila satu data diakses secara bersama-sama oleh lebih dari satu pemakai pada saat yang bersamaan.

5. *Data Dictionary*

DBMS harus menyediakan data *dictionary*.