

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Konsep Dasar Web**

##### **2.1.1. Internet**

Internet merupakan perpustakaan multimedia yang sangat lengkap, bahkan internet dipandang sebagai dunia tersendiri atau dunia maya. Dikatakan sebagai perpustakaan sebab hampir seluruh informasi bidang ilmu pengetahuan disuguhkan internet, seperti bisnis, hiburan, lembaga pemerintahan maupun swasta dari seluruh negara yang ada di dunia. Khususnya untuk bidang pendidikan, pengguna dapat memperoleh informasi mengenai universitas/institusi/akademik, lembaga-lembaga pendidikan, museum, perpustakaan, data bibliografi, seperti buku, jurnal, disertasi dan lain-lain. Informasi mutakhir seperti surat kabar disetiap negara untuk setiap harinya, serta seminar dan pertemuan ilmiah sejenis yang sedang berlangsung dan yang akan diselenggarakan, kemudian yang terpenting lagi adalah bahwa informasi di internet setiap hari bertambah terus dengan jumlah cukup banyak dan informasi yang disediakan selalu *up to date*.

Menurut Supriyanto (2008:60) menyimpulkan bahwa:

Internet ialah hubungan antara berbagai jenis komputer dan jaringan di dunia yang berbeda sistem operasi maupun aplikasinya dimana hubungan tersebut memanfaatkan kemajuan komunikasi (telepon dan satelit) yang menggunakan protokol standar dalam berkomunikasi yaitu protokol TCP/IP atau *Transmission Control/Internet Protocol*.

Pengertian internet diatas dimaknai sebagai media komunikasi dan informasi modern yang dapat dimanfaatkan secara global oleh pengguna diseluruh

dunia dalam interkoneksi antar jaringan komputer yang terbentuk melalui sarana berupa penyedia akses (*provider*) internet, sehingga internet sebagai media informasi dapat menjadi sarana yang efektif dan efisien untuk melakukan pertukaran dan penyebaran informasi tanpa terhalang oleh jarak, perbedaan waktu dan juga faktor geografis bagi siapa saja yang ingin mengakses informasi tersebut.

### 2.1.2. *Website*

*Website* awalnya merupakan suatu layanan sajian informasi yang menggunakan konsep *hyperlink*, yang memudahkan *surfer* atau pengguna internet melakukan penelusuran informasi di internet. Informasi yang disajikan dengan web menggunakan konsep multimedia, informasi dapat disajikan dengan menggunakan banyak media, seperti teks, gambar, animasi, suara, atau film.

Menurut Hidayat (2010:6), “*website* adalah keseluruhan halaman-halaman web yang terdapat dalam sebuah domain yang mengandung informasi. Sebuah *website* biasanya dibangun atas banyak halaman web yang saling berhubungan”. Jadi dapat dikatakan bahwa, pengertian *website* adalah kumpulan halaman-halaman. yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara, dan atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing dihubungkan dengan jaringan-jaringan halaman. Hubungan antara satu halaman *website* dengan halaman *website* lainnya disebut dengan *hyperlink*, sedangkan teks yang dijadikan media penghubung disebut *hypertext*.

Berdasarkan sifatnya, *website* dibedakan ke dalam 2 jenis yaitu menurut Hidayat (2010:8):

a. *Website Statis*

*Website* statis merupakan sebuah *website* yang kontennya jarang diubah, hal ini dikarenakan *website* ini memang tidak ditujukan untuk selalu di *update* sesering mungkin, adapun contoh dari *website* ini adalah *website* profil sebuah organisasi dan *website* profil perusahaan.

b. *Website Dinamis*

Berbeda dengan *website* statis, *website* dinamis merupakan sebuah *website* dengan konten yang selalu diubah setiap saat. Ada banyak *website* dinamis yang satu ini, di antaranya yaitu seperti blog, wikipedia, dan masih banyak lagi yang lainnya.

Selain berdasarkan sifatnya, *website* juga dibedakan berdasarkan fungsinya dan yang biasa digunakan antara lain :

a. *Search Engine* atau Mesin Pencarian

*Search Engine* atau mesin pencari adalah jenis *website* yang khusus mengumpulkan daftar *website* yang bisa ditemukan di internet dalam database kemudian menampilkan daftar indeks ini berdasarkan kata kunci atau “*keywords*” yang dicari oleh *user*. Mayoritas pengguna internet sudah familiar dengan situs *search engine* untuk mencari informasi yang mereka butuhkan melalui internet. Beberapa contoh *search engine* antara lain : *Google Search*, *Bing*, *Yahoo Search* dan lain-lain.

b. Web Portal

Web Portal adalah situs yang mengumpulkan dan menyediakan aneka informasi dari berbagai sumber untuk ditampilkan kepada *user*, jika *user* tertarik untuk mengetahui informasi yang ada dengan lebih lengkap, *user* akan diarahkan ke sumber yang aslinya. Namun pada umumnya web portal tidak hanya menampilkan informasi dari sumber luar, kadang mereka juga menampilkan informasi-informasi dalam *website* mereka sendiri. Beberapa contoh web portal antara lain : *Yahoo*, *msn*, *msnbc* dan lain-lain.

c. *Social Networking*/Sosial Media

Social media termasuk salah satu fenomena yang luar biasa hingga saat ini, khususnya diwakili oleh *Facebook*. Situs *social networking* atau situs pertemanan *online* adalah situs yang disediakan bagi para membeinya untuk bertukar informasi dan media elektronik lainnya seperti foto, musik dan video. Beberapa situs *social networking* yang terkenal sebelum fenomena *Facebook* dan *Twitter* antara lain *Multiply*, *Friendster* dan *MySpace*. Kemudian bermunculan pula sosial media dengan tema-tema yang lebih spesifik seperti sosial media untuk berbagi lokasi (*FourSquare*), sosial media *entertainment* (*GetGlue*) dan masih banyak lagi.

d. Forum

Forum termasuk salah satu jenis *website* yang pernah menjamur di Indonesia. Apalagi sejak fenomena KasKus sebagai forum no 1 dan juga *website* dengan pengunjung terbanyak di Indonesia. Forum

dibuat untuk ajang diskusi dan tukar informasi. Ada yang spesifik ada juga yang bersifat umum, dan dalam forum ini terdapat sub-sub forum berdasarkan kategori-kategori tertentu yang ditetapkan oleh admin. Untuk membangun sebuah forum yang lengkap membutuhkan waktu yang sangat banyak. Untungnya sudah tersedia *software-software* pembuat forum yang siap pakai, baik yang gratis ataupun berbayar. *Software* pembuat forum yang terkenal adalah vBulletin, yang digunakan oleh kaskus dan banyak forum-forum internet lainnya. Sedangkan beberapa forum gratisan dan open-source antara lain : yabb, vanilla, SMF, phpbb dan lain-lain. Sedangkan contoh *website* forum antara lain Kaskus, Modifikasi, detikForum dan lain-lain.

e. Portal Berita dan Informasi

Portal berita dan informasi adalah situs yang memuat berita-berita dan artikel-artikel untuk dibaca pengunjung. Bisa juga berisi opini dan komentar-komentar seputar politik, teknologi dan lain-lain. Situs-situs berita ini seperti layaknya koran, majalah dan buletin *online*. Konsep dan karakteristik situs berita berbeda-beda sesuai dengan jenis berita dan target pembacanya. Beberapa contoh situs berita adalah detik, kompas, cosmopolitan dan lain-lain.

f. *Company Profile*

*Company profile* biasanya berupa *website* sederhana untuk menampilkan profil dan produk atau jasa serta portfolio sebuah perusahaan. *Website* jenis ini hanya untuk memperkenalkan profil perusahaan meskipun juga dapat menjadi alat promosi jika dikelola

dengan benar. Selain profil, produk dan portfolio berupa konten teks dan foto atau video, fitur form kontak termasuk yang penting dalam *website* ini agar pengunjung dapat dengan mudah menghubungi perusahaan pemilik *website* jika mereka memerlukan produk atau jasa perusahaan tersebut. Namun belakangan ini situs *Company Profile* semakin berkembang dengan menambahkan fitur-fitur tertentu seperti artikel-artikel terkait, bahkan forum diskusi. Tujuannya adalah untuk memperkenalkan bisnis melalui dunia maya.

g. *Online Store (Ecommerce)*

*Online store* atau toko *online* sudah jelas peruntukannya adalah untuk menjual barang secara *online*. Fitur-fitur utama *online store* adalah katalog produk dengan informasi yang lengkap serta *shopping cart*, yaitu fitur untuk pengunjung untuk melakukan pemesanan produk secara *online*. *Online store* bisa bermacam-macam bentuk dan konsepnya. Fitur-fitur *online store* juga bisa bermacam-macam disesuaikan dengan *budget* dan kemudahan yang didapatkan. Baik kemudahan mengelola *online store* itu sendiri ataupun kemudahan bagi pengunjung untuk memilih dan berbelanja produk. Menjamurnya *online store* di internet memicu munculnya banyak *software open source* untuk membangun *online store*. Dengan *software-software open-source*, siapa saja dapat membangun *online store* nya sendiri jika tidak memiliki *budget* untuk membayar jasa pembuatan *website* profesional. Namun membangun *online store* dengan menggunakan *software open source* juga bukan hal mudah. Diperlukan waktu untuk

menguasai penggunaan *software*. Belum lagi harus mengerti tentang cara *hosting* dan mengatur konfigurasi *hosting*. Dan yang paling penting adalah bagaimana membuat tampilan *website open source* menjadi menarik dan profesional. Yang sangat disayangkan, banyak jasa pembuatan *website* yang membangun toko *online* klien mereka dengan menggunakan *open source* seperti *oscommerce*, *joomla* dan *wordpress*. Jasa-jasa seperti ini relatif lebih murah, namun klien pemilik *website* baru akan menyadari belakangan kesulitan-kesulitan mereka dalam mengelola toko *online* tersebut karena konfigurasi CMS atau admin area toko *online* yang dihasilkan dari *software open source* sangatlah rumit dan kompleks. Kelemahan lainnya, biasanya *software open source* harus di *update* secara rutin karena sifatnya yang terbuka sehingga mudah dicari kelemahannya dan gampang di bobol oleh *hacker*. *Software online store open source* memang disediakan bagi mereka yang ingin membangun toko *online* nya sendiri dan mengelola sendiri.

### **2.1.3. Bahasa Pemrograman**

Bahasa pemrograman (*programming language*) adalah sebuah instruksi standar untuk memerintah komputer agar mempunyai fungsi tertentu. Bahasa ini memungkinkan seseorang dapat menentukan secara persis data mana yang akan diolah oleh komputer, bagaimana data ini akan disimpan atau diteruskan dan jenis langkah apa secara persis yang akan diambil dalam berbagai situasi. Fungsi bahasa pemrograman yaitu memerintah komputer untuk mengolah data sesuai

dengan alur berpikir yang kita inginkan. Keluaran dari bahasa pemrograman tersebut berupa program atau aplikasi.

Dalam pembuatan sebuah *website*, bisa menggunakan beberapa bahasa pemrograman sebagai dasar dalam pembuatan web dan desain web. Berikut bahasa pemrograman yang biasa digunakan dalam pembuatan web, antara lain:

1. HTML (*Hypertext Markup Language*)

*HyperText Markup Language* (HTML) adalah sebuah bahasa pemrograman yang digunakan untuk membuat sebuah halaman web dan menampilkan berbagai informasi di dalam sebuah *browser* internet. HTML saat ini merupakan standar Internet yang didefinisikan dan dikendalikan penggunaannya oleh *World Wide Web*. HTML berupa kode-kode tag yang menginstruksikan *browser* untuk menghasilkan tampilan sesuai dengan yang diinginkan, HTML dapat dibuka dengan menggunakan *browser* web seperti *Mozilla Firefox*, *Internet Explorer* Atau *Google Chrome*.

Menurut Arief (2011:23) menyimpulkan bahwa:

HTML merupakan salah satu format yang digunakan dalam pembuatan dokumen atau aplikasi yang berjalan di halaman web. HTML merupakan bahasa pemrograman fleksibel dimana kita bisa meletakkan *script* dari bahasa pemrograman lainnya, seperti JAVA, VB, C, dan lainnya.

*Hypertext* dalam HTML berarti bahwa kita dapat menuju ke suatu tempat, misal *website* atau halaman *homepage* lain, dengan cara memilih suatu *link* yang biasanya digaris bawahi atau diwakili oleh suatu gambar. Selain *link* ke *website* atau *homepage* halaman lain, *hypertext* ini juga mengizinkan kita untuk menuju ke salah satu bagian dalam satu teks itu sendiri. HTML tidak berdiri sendiri, agar ia dapat bertugas dalam membangun halaman

web, ia harus ditulis dalam *software* atau aplikasi tertentu, yang dikenal sebagai *HTML Editor*. *HTML Editor* inilah yang bertugas untuk menerjemahkan bahasa HTML menjadi halaman web yang siap dilihat oleh para *surfer* di seluruh dunia.

Secara umum, ada dua jenis *HTML Editor*, yaitu:

a. *Text Editor*

*Text Editor* biasa digunakan oleh mereka yang sudah mahir dalam menggunakan bahasa HTML, karena melalui *editor* jenis ini anda dapat langsung menuliskan kode-kode HTML satu persatu, sesuai prosedur teknis yang berlaku. Untuk *editor* jenis ini, kita dapat menggunakan *notepad*.

b. *WYSIWYG Editor*

*WYSIWYG Editor* adalah solusi bagi mereka yang belum mahir dalam menggunakan bahasa HTML. Pada jenis aplikasi ini, kita dapat membangun halaman web dengan lebih mudah, karena apa yang terlihat di layar akan sama dengan hasil yang anda dapatkan. *WYSIWYG* adalah singkatan dari *What You See Is What You Get*. Untuk *editor* jenis ini, kita dapat menggunakan aplikasi *Microsoft Word*, *Excel*, *Access*, *PowerPoint*, *Outlook*, *FrontPage* dan yang cukup populer *Macromedia Dreamweaver*.

```
<html>
<head>
<title>tempat untuk menempatkan judul halaman web
</title>
</head>
<body>tempat untuk menempatkan informasi
</body>
</html>
```

## Gambar II.1. Contoh *Script* HTML

### 2. PHP

PHP adalah singkatan dari *Hypertext Preprocessor*, yaitu bahasa pemrograman yang digunakan secara luas untuk penanganan pembuatan dan pengembangan sebuah situs web dan bisa digunakan bersamaan dengan HTML.

Menurut Kadir (2009:246) mengemukakan bahwa: PHP adalah *Script* yang dijalankan di server. Jadi, konsepnya berbeda dengan *JavaScript* yang dijalankan pada sisi klien. Keuntungan penggunaan PHP, kode yang menyusun program tidak perlu dibagikan ke pemakai, yang berarti bahwa kerahasiaan kode dapat dilindungi.

PHP disebut juga pemrograman *Server Side Programming*, hal ini dikarenakan seluruh prosesnya dijalankan pada server. PHP adalah suatu bahasa dengan hak cipta terbuka atau yang juga dikenal dengan *open source* yaitu pengguna data mengembangkan kode-kode fungsi sesuai kebutuhannya. Cara penulisan *script* PHP dibedakan menjadi 2 yaitu:

#### a. *Embedded Script*

*Script* yang dimaksud dari *embedded script* adalah *script* PHP yang disisipkan diantara tag-tag HTML. *Script* PHP digunakan apabila isi dari suatu dokumen HTML diinginkan dari hasil eksekusi suatu *script* PHP, selama informasi masih tidak membutuhkan program maka pemrogram umumnya tidak akan menggunakan program.

```

<html>
<head>
<title>
Contoh
</title>
</head>
<body>
<?php
echo "hai, saya dari script php!";
?>
</body>
</html>

```

**Gambar II.2. Contoh *Embedded Script***

b. *Non Embedded Script*

Script PHP pada cara ini digunakan sebagai murni pembuatan program PHP, tag HTML yang dihasilkan untuk membuat dokumen merupakan bagian dari *script* PHP.

```

<?php
echo "<html>";
echo "<head>";
echo "<title>";
echo "contoh 02-Pure On the Fly";
echo "</title>";
echo "</head>";
echo "<body>";
echo "<p>Teks dokumen yang dihasilkan dengan
menggunakan script PHP </p>";
echo "</body>";
echo "</html>";
?>

```

**Gambar II.3. Contoh *Non Embedded Script***

3. CSS (*Cascading Style Sheet*)

*Cascading Style Sheet* (CSS) merupakan salah satu bahasa pemrograman web untuk mengendalikan beberapa komponen dalam sebuah web sehingga akan lebih terstruktur dan seragam. Sama halnya *styles* dalam

aplikasi pengolahan kata seperti *Microsoft Word* yang dapat ,mengukur beberapa *style*, misalnya *heading*, *subbab*, *bodytext*, *footer*, *images*, dan *style* lainnya untuk dapat digunakan bersama-sama dalam beberapa berkas (*file*). Menurut Kadir (2009:450) menyimpulkan bahwa “CSS (*Cascading Style Sheet*) digunakan dalam dokumen HTML untuk menciptakan suatu *style* (penyajian) yang dapat membuat kemampuan HTML menjadi lebih luas”. CSS dapat mengendalikan ukuran gambar, warna bagian tubuh pada teks, warna tabel, ukuran *border*, warna *border*, warna *hyperlink*, warna *mouse over*, spasi antar paragraf, spasi antar teks, margin atas, kiri, kanan, bawah, dan perimeter lainnya.

```
<style>
#header {
    height: 155px;
    width:auto;
    background:#66F url(Adidas.jpg) no-repeat;
    position: relative;
    color: white;
    font-family:"Times New Roman", Times, serif ;
    border: 0px solid #000;
}
</style>
```

**Gambar II.4. Contoh *Cascading Style Sheet***

#### 4. *JavaScript*

Dalam pembuatan sebuah halaman web, Anda juga memerlukan pemrograman *JavaScript* agar web yang dibuat lebih dinamis. Biasanya *JavaScript* di gunakan untuk memberikan fungsi, komunikasi dengan server dan validasi (seperti tanggal, *email*, *password*, dan lain sebagainya). *JavaScript* sendiri adalah bahasa *script* untuk web yang dapat bekerja di sebagian besar *browser* ternama seperti *internet explorer*, *firefox*, dan

*chrome*. *Java* yang merupakan *oriented object programming*, sedangkan *script* merupakan serangkaian intruksi di dalam program. Menurut Kadir (2009:8) “*JavaScript* adalah bahas skrip (bahasa yang kodenya ditulis menggunakan teks biasa) yang ditempelkan pada dokumen HTML dan diproses pada sisi klien.”

Beberapa contoh yang bisa dilakukan melalui *Javacript* menurut Kadir (2009:8):

- a. Menampilkan jam lokal pada halaman web
- b. Mengatur warna latar belakang halaman web
- c. Mengganti gambar ketika pemakai menempatkan penunjuk *mouse* ke suatu gambar
- d. Memvalidasi keabsahan data yang dimasukkan oleh pemakai
- e. Menandai semua data yang akan dihapus dengan cara memberi tanda centang pada kotak cek. Hal seperti ini biasa disediakan pada pembaca *e-mail* yang berbasis web

## 5. *jQuery*

*jQuery* merupakan suatu *framework (library) Javascript* yang menekankan bagaimana interaksi antara *Javascript* dan HTML. *jQuery* pertama kali dirilis pada tahun 2006 oleh John Resig. Menurut Flanagan (2011:1) *jQuery* merupakan *library* dari *JavaScript* yang fokus dalam meng-*query* objek *JavaScript*. *Selector* yang digunakan dalam *jQuery* biasanya mengakses kelas CSS untuk menangkap DOM dan mengolahnya dengan *method* tertentu.

6. *XML (Extensible Markup Language)*

Pengertian dari XML (*Extensible Markup Language*) adalah sebuah bahasa penanda yang disyorkan oleh Konsortium Jaringan Sejagat (W3C) untuk kegunaan umum dalam format yang boleh dibaca manusia dan boleh dibaca mesin. Ia didefinisikan dalam XML 1.0 *Specification* yang dikeluarkan W3C, dan beberapa spesifikasi berkaitan yang lain Tepatnya XML yaitu kelanjutan dari HTML (*HyperText Markup Language*) dimana yang merupakan bahasa standar untuk melacak Internet. Jadi, untuk membaca bahasa markup ada kesinambungannya.

Perbedaan antara XML dan HTML menurut Sidik dan Pohan (2007:344):

- a. XML didesain untuk membawa data
- b. XML tidak menggantikan HTML
- c. XML dan HTML didesain untuk tujuan yang berbeda
- d. XML didesain untuk mendeskripsikan data dan difokuskan pada apa sebenarnya data itu
- e. HTML didesain untuk menampilkan data dan bagaimana data harus terlihat
- f. HTML adalah bagaimana menampilkan informasi, XML adalah bagaimana mendeskripsikan informasi

#### 2.1.4. Basis Data

##### 1. Pengertian Basis Data (*Database*)

Menurut Kustiyaningsih (2011:146), “*Database* adalah Struktur penyimpanan data. Untuk menambah, mengakses dan memproses data yang disimpan dalam sebuah *database* komputer, diperlukan sistem manajemen *database* seperti *MYSQL Server*.”

Menurut Bentley dan Whitten (2007:520-522) mengemukakan bahwa:

Didalam *database* terdapat data yang disimpan yang terdiri dari struktur *file* dan tabel. *File* sendiri terdiri dari *fields* dan *records*, sedangkan tabel terdiri dari baris dan kolom. *Fields* adalah unit terkecil dari data yang disimpan dalam sebuah *file* atau *database*. Sedangkan *Records* adalah sebuah kumpulan *fields* yang diatur dalam format yang telah ditentukan.

Ada 4 macam tipe *fields* yang terdapat dalam *database*, yaitu:

##### a. *Primary Key*

*Primary Key* adalah sebuah *field* yang mengidentifikasi nilai yang unik dan hanya terdapat satu *record* dalam sebuah *database*.

##### b. *Secondary Key*

*Secondary Key* adalah sebuah *field* yang mengidentifikasi sebuah *record* tunggal atau identifikasi alternatif dalam sebuah *database*.

##### c. *Foreign Key*

*Foreign Key* adalah sebuah *field* yang menunjuk ke *record file* yang berbeda dalam sebuah *database*.

##### d. *Descriptive field*

*Descriptive field* adalah semua *field* lainnya yang tidak menjadi *key* (*non key*) dalam sebuah *database*.

## 2. *Tools* Perangkat Lunak

### a. *MySQL*

Menurut Arief (2011:152), “*MySQL* adalah salah satu jenis *database server* yang sangat terkenal dan banyak digunakan untuk membangun aplikasi web yang menggunakan database sebagai sumber dan pengolahan datanya.” *MySQL* merupakan *database* yang pertama kali didukung oleh bahasa pemrograman *script* untuk internet (PHP dan *Perl*). *MySQL* dan PHP dianggap sebagai pasangan *software* pembangun aplikasi web yang ideal. *MySQL* lebih sering digunakan untuk membangun aplikasi berbasis web, umumnya pengembangan aplikasinya menggunakan bahasa pemrograman *script* PHP. Berikut fungsi-fungsi PHP-*MySQL*:

#### 1) *MySQL\_connect()*

Digunakan untuk melakukan uji dan koneksi kepada *server database MySQL*.

Sintaks:

```
$conn=MySQL_connect("host","username","password")
```

#### 2) *MySQL\_select\_db()*

Digunakan untuk melakukan koneksi kepada *database* yang dalam *server* yang berhasil dikoneksi dengan perintah *MySQL\_connect()*.

Sintaks:

```
$pilih=MySQL_select_db("namadatabase",$conn)
```

3) `my_sql_query()`

Digunakan untuk melakukan eksekusi perintah SQL untuk memanipulasi *database* yang berhasil dilakukan koneksinya dengan menggunakan `MySQL_select_db()`.

Sintaks:

```
$hasil=MySQL_query("SQLStatement")
```

\$hasil akan berupa *record set* apabila *SQLStatement* berupa perintah *SELECT*.

4) `MySQL_fetch_array()`

Digunakan untuk melakukan pemrosesan hasil *query* yang dilakukan dengan perintah `MySQL_query()`.

Sintaks:

```
$array=MySQL_fetch_array($hasil)
```

\$array adalah *array* satu *record* dari *record* \$hasil yang diproses.

Nomor *record* sesuai dengan nomor urut dari proses `MySQL_fetch array` yang sedang dilakukan. \$hasil adalah *record set* yang akan diproses.

b. *SQL*

Menurut Sukamto dan Shalahuddin (2014:46) “SQL (*Structured Query Language*) adalah bahasa yang digunakan untuk mengelola data pada RDBMS”. SQL merupakan bahasa standar yang digunakan untuk manipulasi dan memperoleh data dari sebuah basis data relasional. Perintah SQL dikelompokkan menjadi dua macam, yaitu:

1) *Data Definition Language (DDL)*

Adalah perintah SQL yang digunakan untuk menjelaskan objek dari basis data, dengan kata lain DDL digunakan untuk membuat kerangka basis data. Perintah yang digunakan pada DDL:

- a) *CREATE* : Digunakan untuk membuat objek database
- b) *ALTER* : Digunakan untuk memodifikasi objek database
- c) *DROP* : Digunakan untuk menghapus objek database

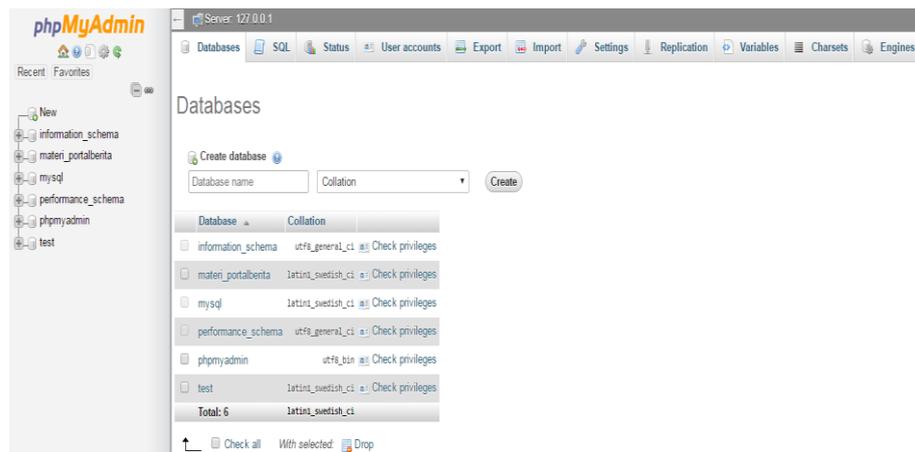
2) *Data Manipulation Language (DML)*

Adalah perintah yang digunakan untuk mengoperasikan atau memanipulasi isi basis data. Perintah yang digunakan pada DML:

- a) *CREATE* : Digunakan untuk membuat objek *database*
- b) *INSERT* : Digunakan untuk menambahkan data ke dalam basis data
- c) *DELETE* : Digunakan untuk menghapus data pada *database*
- d) *UPDATE* : Digunakan untuk memodifikasi data pada basis data

c. *PHP MyAdmin*

Menurut Sibero (2011:376), “*phpMyAdmin* adalah aplikasi web yang dibuat oleh *phpmyadmin.net*, *phpMyAdmin* digunakan untuk administrasi database *MySQL*.” Dengan aplikasi *phpMyAdmin* kita bias mengelola *database* sebagai *root* atau juga sebagai *user* biasa, kita bias membuat *database* baru, mengelola *database* dan melakukan operasi perintah-perintah *database* secara lengkap seperti saat kita di *MySQL Prompt*.



**Gambar II.5. Tampilan PHP MyAdmin**

d. *XAMPP*

*XAMPP* adalah *software* web *server apache* yang di dalamnya tertanam *server MySQL* yang didukung dengan bahasa pemrograman *PHP* untuk membuat *website* yang dinamis.

Menurut Riyanto (2010:1) mengemukakan bahwa: *XAMPP* merupakan paket *PHP* dan *MySQL* berbasis *open source*, yang dapat digunakan sebagai *tool* pembantu pengembangan aplikasi berbasis *PHP*. *XAMPP* mengombinasikan beberapa paket perangkat lunak berbeda ke dalam satu paket. Memanfaatkan *XAMPP* sebagai *database* karena *XAMPP* menyediakan aplikasi *database MySQL* dengan *interface* lebih mudah dalam pengoperasiannya, *tool-tool* yang disediakan cukup lengkap dan memenuhi kebutuhan perancangan data base selain itu *XAMPP* aplikasi gratis.

e. *Apache*

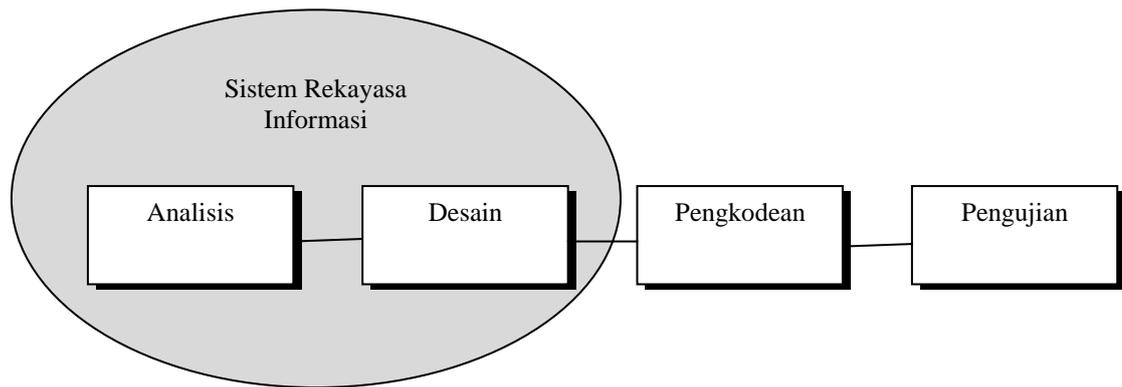
Menurut Kurniawan (2008:2) ”*Apache* adalah web *server* yang dapat dijalankan dibanyak sistem operasi (Unix, BSD, Linux, *Microsoft windows* dan *Novell Netware* serta *Platfrom* lainnya) yang berguna untuk memfungsikan situs web”. Protokol yang digunakan untuk melayani fasilitas web ini menggunakan *HTTP*.

f. *Macromedia Dreamweaver 8*

*Macromedia Dreamweaver8* yang diciptakan pada tahun 1997, merupakan program aplikasi web *editor* yang cukup populer untuk pembuatan *website*, selain tampilannya yang menarik, kemampuan fitur-fiturnya juga memudahkan para pemakainnya untuk menciptakan halaman web. Namun sekarang *Macromedia* dimiliki oleh *adobe* yang mengakuisi pada tahun 2005, dan berganti nama menjadi *Adobe Dreamweaver*. *Adobe dreamweaver* adalah *script editor* profesional yang digunakan untuk melakukan desain secara visual dan mengelola halaman web serta pembuat aplikasi web. Pengembang dan *programmer* aplikasi web lebih menyukai menggunakan *dreamweaver*, karena *dreamweaver* berbasiskan WYSIWYG (*What You See Is What You Get*). Menurut Madcoms (2011:2) memberikan batasan bahwa “*dreamweaver* adalah sebuah HTML *editor* profesional untuk mendesain web secara visual dan mengelola situs atau halaman web”.

### **2.1.5. Model Pengembangan Perangkat Lunak**

Menurut Sukamto dan Shalahuddin (2013:28), “Model SDLC (*Software Development Life Cycle*) air terjun (*Waterfall*) sering juga disebut model sekuensi linier (*sequential linier*) atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis desain, pengodean, pengujian, dan tahap pendukung (*support*).” Berikut gambar model air terjun:



Sumber: Sukamto dan Shalahuddin (2013:29)

### **Gambar II.6. Ilustrasi Model *Waterfall***

1. Analisa kebutuhan perangkat lunak

Proses pengumpulan kebutuhan dilakukan secara intensif untuk memesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*. Spesifikasi kebutuhan perannkat lunak pada tahap ini perlu untuk didokumentasikan.

2. Desain

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengkodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikan.

3. Pembuatan kode program

Desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

#### 4. Pengujian

Pengujian fokus pada perangkat lunak secara dari segi logik (*logic*), fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

#### 5. Pendukung (*support*) atau pemeliharaan (*maintenance*)

Tidak menutup kemungkinan sebuah perangkat lunak harus mengalami perubahan ketika sudah dikirimkan ke *user*. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap pendukung atau pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru.

### **2.1.6. Sistem Informasi Pemesanan**

Suatu sistem pada dasarnya adalah sekelompok unsur yang erat berhubungan satu dengan lainnya, yang berfungsi bersama-sama untuk mencapai tujuan tertentu. Menurut Kristanto (2008:1) “Sistem merupakan jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran tertentu”. Sedangkan Menurut Yakub (2012:1), “Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang berhubungan, terkumpul bersama-sama untuk melakukan suatu kegiatan atau tujuan tertentu”.

Dari penjelasan di atas maka dapat disimpulkan bahwa pengertian sistem adalah suatu bagian yang saling berinteraksi dan berhubungan dengan maksud untuk mencapai tujuan atau sasaran yang sudah ditetapkan bersama.

Informasi juga merupakan data yang sudah dibentuk menjadi bentuk yang berarti dan berguna bagi manusia. Jadi yang dimaksud dengan informasi adalah data sederhana yang telah dibentuk sehingga mempunyai arti dan berguna di dalam keadaan tertentu. Menurut Sutarman (2012:14) “Informasi adalah sekumpulan fakta (data) yang diorganisasikan dengan cara tertentu sehingga mereka mempunyai arti bagi si penerima”.

Dengan demikian sistem informasi adalah sekumpulan komponen yang saling berhubungan yang mengumpulkan (mendapatkan kembali), memproses, menyimpan, dan mendistribusikan informasi untuk mendukung pengambilan keputusan dan pengendalian dalam organisasi.

Pemesanan adalah salah satu prosedur yang dilakukan pelanggan untuk memperoleh tiket. Menurut Kamus Besar Bahasa Indonesia yang dimaksud pemesanan adalah “proses, perbuatan, cara memesan (tempat, barang, dsb) kepada orang lain”.

## **2.2. Teori Pendukung**

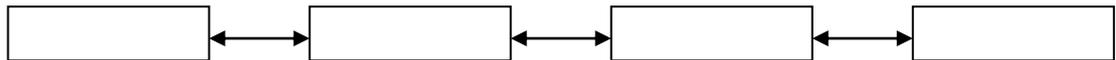
### **2.2.1. Struktur Navigasi**

Menurut Binanto (2010:268) “struktur navigasi adalah gabungan dari struktur referensi informasi situs web dan mekanisme *link* yang mendukung pengunjung untuk melakukan penjelajahan situs”.

Bentuk dasar dari struktur navigasi yang biasa digunakan dalam proses pembuatan aplikasi multimedia ada empat macam, yaitu struktur navigasi *linier*, *hirarki*, *non linier* dan campuran.

### 1. Struktur Navigasi *Linier*

Struktur navigasi *linier* hanya mempunyai satu rangkaian cerita yang berurutan yang menampilkan satu demi satu tampilan layar secara berurut menurut urutannya. Tampilan yang dapat ditampilkan pada struktur jenis ini adalah satu halaman sebelumnya atau satu halaman sesudahnya, tidak dapat dua halaman sebelumnya atau dua halaman sesudahnya, pengguna akan melakukan navigasi secara berurutan, dalam *frame* atau *byte* informasi satu ke yang lainnya.

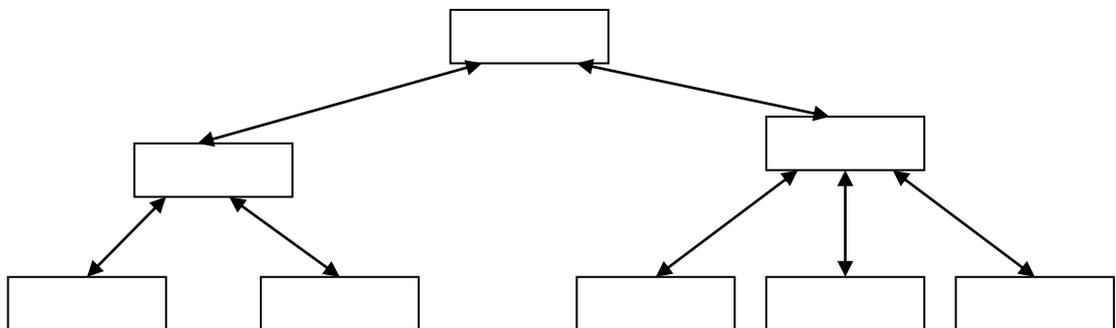


Sumber : Binanto (2010:269)

**Gambar II.7. Struktur Navigasi Linier**

### 2. Struktur Navigasi Hirarki

Struktur dasar ini disebut juga struktur *linier* dengan percabangan karena pengguna melakukan navigasi disepanjang cabang pohon struktur yang terbentuk oleh logika isi.

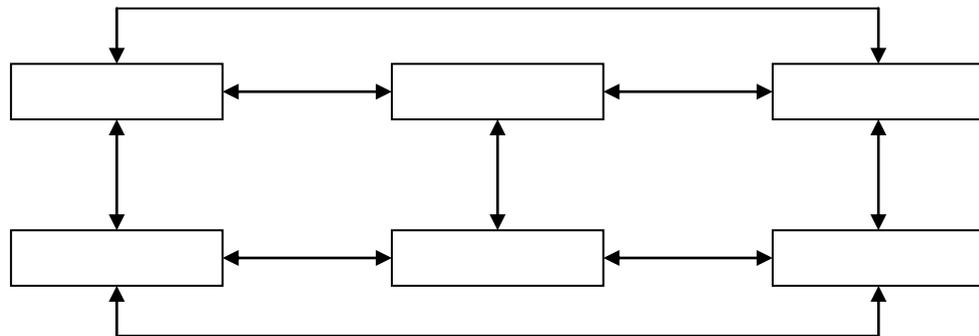


Sumber : Binanto (2010:269)

**Gambar II.8. Struktur Navigasi Hirarki**

### 3. Struktur Navigasi Tidak Berurut (*non Linier*)

Struktur navigasi *non linier* (tidak terurut) merupakan pengembangan dari struktur navigasi *linier*. Pada struktur ini diperkenankan membuat navigasi bercabang. Percabangan yang dibuat pada struktur *non-linier* ini berbeda dengan percabangan pada struktur hirarki, karena pada percabangan *non-linier* ini walaupun terdapat percabangan pada tiap-tiap tampilan mempunyai kedudukan yang sama yaitu tidak ada *Master Page* dan *Slave Page*, pengguna akan melakukan navigasi dengan bebas melalui isi proyek dengan tidak terikat dengan jalur yang sudah ditentukan.

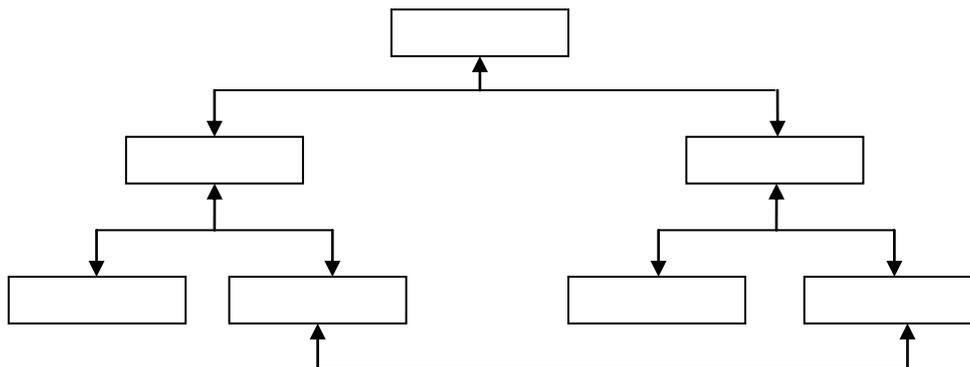


Sumber : Binanto (2010:269)

**Gambar II.9. Struktur Navigasi Non Linier**

### 4. Struktur Navigasi Campuran (*Composite*)

Pengguna akan melakukan navigasi dengan bebas (*secara non linear*), tetapi terkadang dibatasi presentasi linier film atau informasi penting dan atau pada data yang paling terorganisasi secara logis pada suatu hirarki.



Sumber : Binanto (2010:260)

**Gambar II.10. Struktur Navigasi Campuran (*Composite*)**

### 2.2.2. *Enterprise Relationship Diagram (ERD)*

#### 1. Definisi *Enterprise Relationship Diagram (ERD)*

Menurut Sutanta (2011:91) “*Enterprise Relationship Diagram (ERD)* merupakan suatu model data yang dikembangkan berdasarkan objek”. *Enterprise Relationship Diagram (ERD)* digunakan untuk menjelaskan hubungan antar data dalam basis data kepada pengguna secara logis. *Enterprise Relationship Diagram (ERD)* di dasarkan pada suatu persepsi bahwa real world terdiri atas obyek-obyek dasar tersebut. Penggunaan *Enterprise Relationship Diagram (ERD)* relatif mudah dipahami, bahkan oleh para pengguna yang awam. Bagi perancang atau analis sistem, *Enterprise Relationship Diagram (ERD)* berguna untuk memodelkan sistem yang nantinya basis data akan kembangkan. Model ini juga membantu perancang atau analis sistem pada saat melakukan analis dan perancangan basis data karena model ini dapat menunjukkan macam data yang dibutuhkan dan kerelasian antar data di dalamnya.

Menurut Sukanto dan Shalahuddin (2014:50) mengatakan bahwa “Pemodelan awal basis data yang paling banyak digunakan adalah menggunakan *Entity Relationship Diagram (ERD)*”.

## 2. Komponen *Entity Relationship Diagram (ERD)*

Adapun komponen yang terdapat pada *Entity Relationship Diagram (ERD)* sebagai berikut:

- a. Entitas (*Entity*) adalah sesuatu yang nyata atau abstrak dimana kita akan menyimpan data.
- b. Relasi (*Realtionship*) adalah Hubungan alamiah yang terjadi antara satu atau lebih entitas.
- c. Atribut (*Attribute*) adalah Ciri umum semua atau sebagian besar instansi pada entitas tertentu. Sebutan lain dari atribut adalah properti, elemen data, dan *field*.

## 3. Derajat *Relationship (Kardinalitas)*

Menurut Yakub (2008:33) mengemukakan bahwa “Model relasi ini berdasarakan persepsi dunia nyata diantaranya himpunan objek dan diidentifikasi secara unik dan objeknya dapat berbentuk orang, barang dan sebagainya”. Kardinalitas relasi menunjukkan maksimum entitas yang dapat berelasi dengan entitas pada himpunan entitas yang lain. Kardinalitas relasi yang terjadi diantara dua himpunan entitas (misalkan A dan B) dapat berupa satu ke satu (*one to one*), satu ke banyak (*one to many*), banyak ke satu (*many to one*) dan banyak ke banyak (*many to many*).

a. Satu ke Satu (*One To One*)

Berarti setiap entitas pada himpunan entitas A berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas B dan begitu juga sebaliknya setiap entitas pada himpunan entitas B berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas A.

b. Satu ke Banyak (*One To Many*)

Berarti setiap entitas pada himpunan entitas A dan dapat berhubungan dengan banyak entitas pada himpunan entitas B, tetapi tidak sebaliknya, dimana setiap entitas pada himpunan entitas yang berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas A.

c. Banyak ke Satu (*Many To One*)

Berarti setiap entitas pada himpunan entitas A berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas B, tetapi tidak sebaliknya, dimana setiap entitas pada himpunan entitas A berhubungan dengan paling banyak satu entitas pada himpunan entitas B.

d. Banyak ke Banyak (*Many To Many*)

Berarti setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, demikian juga sebaliknya, dimana setiap entitas pada himpunan entitas B dapat berhubungan dengan banyak entitas pada himpunan entitas A.

### 2.2.3. *Logical Record Structure (LRS)*

Menurut Hasugian dan Shidiq (2012:608) memberikan batasan bahwa “LRS adalah sebuah model sistem yang digambarkan dengan sebuah diagram-ER akan mengikuti pola atau aturan permodelan tertentu dalam kaitannya dengan konvensi ke LRS”. Beberapa tipe *record* digambarkan oleh kotak persegi panjang dan dengan nama yang unik. Perbedaan LRS dengan diagram ER adalah nama tipe *record* berada diluar kotak *field* tipe *record* ditempatkan. LRS terdiri dari *link-link* diantara tipe *record*. *Link* ini menunjukkan arah dari satu tipe *record field-field* yang kelihatan pada kedua *link* tipe *record*. Penggambaran LRS mulai dengan menggunakan model yang dimengerti. Dua metode yang dapat digunakan, dimulai dengan hubungan kedua model yang dapat dikonversikan ke LRS, metode yang lain dimulai dengan ER diagram dan langsung dikonversikan ke LRS.

Perubahan yang terjadi yaitu mengikuti aturan-aturan sebagai berikut (Hasugian dan Shidiq, 2012:608):

1. Setiap entitas akan diubah kebentuk kotak
2. Sebuah atribut relasi disatukan dalam sebuah kotak bersama entitas jika hubungan yang terjadi pada diagram-ER 1:M (relasi bersatu dengan *cardinality* M) atau tingkat hubungan 1:1 (relasi bersatu dengan *cardinality* yang paling membutuhkan referensi)
3. Sebuah relasi dipisah dalam sebuah kotak tersendiri (menjadi entitas baru) jika tingkat hubungannya M:M (*many to many*) dan memiliki *foreign key* sebagai *primary key* yang diambil dari kedua entitas yang sebelumnya saling berhubungan

#### 2.2.4. Pengujian Web

Metode pengujian adalah cara atau teknik untuk menguji perangkat lunak, yang mempunyai mekanisme untuk menentukan data uji yang dapat menguji perangkat lunak secara lengkap dan mempunyai kemungkinan tinggi untuk menemukan kesalahan.

Metode pengujian web yang penulis gunakan adalah pengujian kotak hitam (*Black-Box Testing*).

Menurut Rizky (2011:264), berpendapat bahwa:

“*Black-box testing* adalah tipe *testing* yang memperlakukan perangkat lunak yang diketahui kinerja internalnya. Sehingga pada tester memandang perangkat lunak seperti layaknya sebuah “kotak hitam” yang tidak penting dilihat isinya, tapi cukup dikenai proses *testing* dibagian luar.”

Pengujian *black-box* bukan merupakan alternatif dari teknik *white-box*, tetapi merupakan pendekatan komplementer yang kemungkinan besar mampu mengungkap kelas kesalahan daripada metode *white-box*. Pengujian *black-box* berusaha menemukan kesalahan dalam kategori sebagai berikut:

- a. Fungsi-fungsi yang tidak benar atau hilang,
- b. Kesalahan *interface*,
- c. Kesalahan dalam struktur data atau akses *database* eksternal,
- d. Kesalahan kinerja,
- e. Inisialisasi dan kesalahan terminasi.

Tidak seperti pengujian *white-box*, yang dilakukan pada saat awal proses pengujian, pengujian *black-box* cenderung diaplikasikan selama tahap akhir pengujian. Karena pengujian *black-box* memperhatikan struktur kontrol, maka perhatian berfokus pada domain informasi.