

BAB II

LANDASAN TEORI

2.1. Sistem Informasi

Sistem Informasi terdiri dari dua kata, yaitu Sistem dan Informasi. Secara umum, sistem berarti gabungan dari beberapa sub sistem yang bertujuan untuk mencapai satu tujuan. Sedangkan informasi berarti sekumpulan data yang sebelumnya telah diolah sehingga menghasilkan sesuatu yang bermanfaat bagi yang menggunakannya.

Berikut akan dijelaskan lebih detail mengenai apa itu sistem, informasi, dan sistem informasi.

2.1.1. Sistem

Terdapat dua kelompok pendekatan di dalam pendefinisian sistem, yaitu kelompok yang menekankan pada prosedur dan kelompok yang menekankan pada elemen atau komponennya. Pendekatan yang menekankan pada prosedur mendefinisikan sistem sebagai suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu. Sedangkan pendekatan sistem yang lebih menekankan pada elemen atau komponen mendefinisikan sistem sebagai kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. Kedua kelompok definisi ini adalah benar dan tidak bertentangan. Yang berbeda adalah cara pendekatannya. [Sutabri, 2004:2]

Suatu sistem pada dasarnya adalah sekelompok unsur yang erat hubungannya satu dengan yang lain, yang berfungsi bersama-sama untuk mencapai tujuan tertentu. Dari definisi ini, dapat dirinci lebih lanjut pengertian sistem, yaitu sebagai berikut: [Sutabri, 2004:9]

1. Setiap sistem terdiri dari berbagai unsur. Unsur-unsur suatu sistem terdiri dari subsistem yang lebih kecil, yang terdiri pula dari kelompok-kelompok unsur yang membentuk subsistem tersebut.

2. Unsur-unsur tersebut merupakan bagian yang tak terpisahkan dari sistem yang bersangkutan dan berhubungan erat satu sama lain di mana sifat serta kerja sama antarunsur dalam sistem tersebut mempunyai bentuk tertentu.
3. Unsur-unsur di dalam sistem tersebut bekerja sama untuk mencapai tujuan tujuan tertentu.
4. Suatu sistem merupakan bagian dari sistem lain yang lebih besar.

Suatu sistem dibuat untuk menangani sesuatu yang berulang kali atau yang secara rutin terjadi. Pendekatan sistem merupakan suatu filsafat atau persepsi tentang struktur yang mengkoordinasikan kegiatan-kegiatan dan operasi-operasi dalam suatu organisasi dengan cara yang efisien dan yang paling baik. [Sutabri, 2004:10]

2.1.2. Informasi

Informasi merupakan proses lebih lanjut dari data yang sudah memiliki nilai tambah. Informasi dapat dikelompokkan menjadi 3 bagian, yaitu: [Sutabri, 2004:17]

1. Informasi strategis. Informasi ini digunakan untuk mengambil keputusan jangka panjang, mencakup informasi eksternal, rencana perluasan perusahaan, dan sebagainya.
2. Informasi taktis. Informasi ini dibutuhkan untuk mengambil keputusan jangka menengah, seperti informasi penjualan yang dapat dimanfaatkan untuk menyusun rencana penjualan.
3. Informasi teknis. Informasi ini dibutuhkan untuk keperluan operasional sehari-hari, seperti informasi persediaan stok, retur penjualan dan laporan kas harian.

Informasi adalah data yang telah diklasifikasikan atau diolah atau diinterpretasikan untuk digunakan dalam proses pengambilan keputusan. Sistem pengolahan informasi akan mengolah data menjadi informasi atau mengolah data dari bentuk tak berguna menjadi berguna bagi yang menerimanya. Nilai informasi berhubungan dengan keputusan. Bila tidak ada pilihan atau keputusan maka informasi tidak diperlukan. Keputusan dapat berkisar dari keputusan berulang

sederhana sampai keputusan strategis jangka panjang. Nilai informasi dilukiskan paling berarti dalam konteks pengambilan keputusan. [Sutabri, 2004:18]

2.1.3. Sistem Informasi

Dari beberapa pengertian mengenai sistem dan informasi, maka sistem informasi dapat didefinisikan sebagai berikut. "Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan". [Sutabri, 2004:36]

Sistem Informasi terdiri dari komponen-komponen yang disebut dengan istilah building block, yang terdiri dari blok masukan, blok model, blok keluaran, blok teknologi, blok basis data, dan blok kendali. Sebagai suatu sistem, keenam blok tersebut saling berinteraksi satu dengan yang lain membentuk satu kesatuan untuk mencapai sasaran. Adapun penjelasan keenam blok tersebut adalah sebagai berikut: [Sutabri, 2004:36]

1. Blok masukan

Blok masukan mewakili data yang masuk ke dalam sistem informasi. Yang dimaksud dengan masukan di sini termasuk metode dan media untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen-dokumen dasar.

2. Blok model

Blok model ini terdiri dari kombinnasi prosedur, logika, dan model matematik yang akan memanipulasi data input dan data yang tersimpan dari basis data dengan cara yang sudah tertentu untuk menghasilkan keluaran yang diinginkan.

3. Blok keluaran

Produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

4. Blok teknologi

Blok teknologi merupakan *Tool box* dalam sistem informasi. Blok teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian dari sistem secara keseluruhan. Teknologi terdiri dari 3 (tiga) bagian utama yaitu teknisi (*brainware*), perangkat lunak (*software*), dan perangkat keras (*hardware*).

5. Blok basis data

Blok basis data (*Database*) merupakan kumpulan data yang saling berkaitan dan berhubungan satu dengan yang lainnya, tersimpan di perangkat keras komputer dan perangkat lunak yang digunakan untuk memanipulasinya. Data perlu disimpan dalam basis data untuk keperluan penyediaan informasi lebih lanjut. Data di dalam basis data perlu diorganisasikan sedemikian rupa supaya informasi yang baik juga berguna untuk efisiensi kapasitas penyimpanannya. Basis data diakses atau dimanipulasi dengan menggunakan perangkat lunak paket yang disebut dengan DBMS (*Database management system*).

6. Blok kendali

Banyak hal dapat merusak sistem informasi, seperti bencana alam, api, temperatur, air, debu, kecurangan-kecurangan, kegagalan dalam sistem itu sendiri, ketidak-efisienan, sabotase dan sebagainya. Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem tersebut dihindari atau dicegah dan bila terlanjur terjadi maka kesalahan-kesalahan dapat dengan cepat diatasi.

2.2. PHP

PHP merupakan kependekan dari PHP Hypertext Preprocessor yang digunakan sebagai bahasa script *server-side* dalam pengembangan Web yang disisipkan pada dokumen HTML. Penggunaan PHP memungkinkan Web dapat dibuat dinamis sehingga *maintenance* situs Web tersebut menjadi lebih mudah dan efisien. [Peranginangin, 2006:2]

2.2.1. Kelebihan PHP

PHP memiliki banyak kelebihan yang tidak dimiliki oleh bahasa script sejenis. Adapun beberapa kelebihan PHP adalah sebagai berikut: [Peranginangin, 2006:3]

1. PHP difokuskan pada pembuatan script *server-side*, yang bisa melakukan apa saja yang dapat dilakukan oleh CGI, seperti mengumpulkan data dari *Form*, menghasilkan isi halaman Web dinamis, dan kemampuan untuk mengirim serta menerima Cookies, bahkan lebih daripada kemampuan CGI.
2. PHP dapat digunakan pada semua sistem operasi, antara lain Linux, Unix (termasuk variannya HP-UX, Solaris, dan OpenBSD), Microsoft Windows, Mac OS X, RISC OS. PHP juga mendukung banyak Web Server, seperti Apache, Microsoft Internet Information Server (IIS), Personal Web Server (PWS), Netscape dan iPlanet servers, dan sebagainya. Bahkan PHP dapat bekerja sebagai suatu CGI processor.
3. PHP tidak terbatas pada hasil keluaran HTML (Hypertext Markup Language). PHP juga memiliki kemampuan untuk mengolah keluaran gambar, *File* PDF, dan movies Flash. PHP juga dapat menghasilkan teks seperti XHTML dan *File* XML lainnya.
4. Kelebihan lain yang dimiliki oleh PHP adalah dukungannya terhadap banyak *Database* diantaranya adalah Adabas D, dBase, Direct Ms-SQL, Empress, *FilePro*(read-only), FrontBase, Hyperware, IBM DB2, InFormix, Interbase, MySQL, Oracle (OCI7 dan OCI8), PostgreSQL, Sybase.

2.2.2. Tag PHP

Berikut adalah contoh yang umum digunakan untuk menjelaskan tentang PHP sebagai script yang disisipkan dalam dokumen HTML:

```
<html>
<head>
<title>Contoh</title>
</head>
<body>
  <?php
    echo"Hai,saya dari script PHP";
  ?>
</body>
</html>
```

Script diatas merupakan script gabungan antara HTML dan PHP, sedangkan script PHP sendiri terletak diantara *Tag* `<?php` dan `?>`

Saat script ini dijalankan di Browser yang menggunakan Web Server, maka tampilan yang tercetak di layar adalah **Hai, saya dari script PHP**. Hal ini disebabkan fungsi `echo` yang digunakan untuk menampilkan data ke Browser agar bisa diakses oleh *User*. [Sidik, 2004:4]

2.2.3. Tipe data PHP

Secara umum PHP mempunyai tipe data yang sama dengan bahasa pemrograman lainnya. PHP mendukung delapan tipe data yang secara umum terbagi menjadi 3 tipe data, yaitu tipe data *Scalar* (boolean, integer, floating-point, string), tipe data *Compound* (array dan object), dan tipe data khusus (resource dan NULL):

1. Boolean

Tipe data boolean adalah tipe data paling sederhana, yakni untuk menyatakan suatu nilai kebenaran **TRUE** atau **FALSE** yang bersifat case sensitive. [Peranginangin, 2006:19]

2. Integer

Tipe data integer adalah tipe data yang menyatakan bilangan bulat. Jangkauan bilangan integer tergantung pada *platForm*, kira-kira 2 milyar. [Peranginangin, 2006:20]

3. Floating-point

Tipe data floating-point adalah tipe data bilangan **float**, **double**, atau **real** yang menyatakan bilangan pecahan. [Peranginangin, 2006:22]

4. String

Suatu string adalah sekumpulan karakter. Dalam PHP suatu karakter dianggap sebagai suatu byte sehingga ada 256 karakter berbeda. [Peranginangin, 2006:24]

5. Array

Array adalah satu rangkaian elemen bertipe sama yang menempati memori secara teratur yang dapat diacu/diakses secara individu dengan menambahkan

suatu *index* kepada suatu nama unik. Array sering disebut juga sebagai larik. [Peranginangin, 2006:124]

6. Object

Tipe data object adalah tipe data yang memiliki kombinasi struktur data dan beberapa fungsi. Tipe data object pada PHP adalah untuk mendukung pemrograman berorientasi object. [Peranginangin, 2006:29]

7. Resource

Suatu resource adalah suatu variabel khusus sebagai suatu acuan terhadap suatu *external resource*. Resource diciptakan dan digunakan oleh fungsi khusus. [Peranginangin, 2006:29]

8. NULL

Nilai NULL menyatakan bahwa suatu variabel tidak memiliki nilai. NULL hanya merupakan nilai mungkin dari tipe NULL yang telah diperkenalkan pada PHP 4, dan *keyword* NULL adalah case sensitive. [Peranginangin, 2006:29]

2.2.4. Variabel PHP

Variabel adalah suatu lokasi dalam memori komputer untuk menyimpan suatu nilai atau data. Isi variabel bisa berubah-ubah selama proses program. Variabel dalam PHP selalu diawali dengan tanda \$, kemudian diikuti oleh nama variabel yang bersifat case sensitive. [Peranginangin, 2006:34]

Contoh : \$nama="anief";

2.2.5. Operator PHP

Operator adalah simbol atau karakter khusus yang digunakan dalam suatu ekspresi untuk memanipulasi nilai atau variabel dan memberikan suatu hasil. Operator pada PHP dibedakan menjadi operator aritmatika, operator assignment, operator bit, operator relasi, operator ternary, operator kontrol error, operator eksekusi, operator increment/decrement, operator logika, operator string, dan operator array. [Peranginangin, 2006:50]

2.2.6. Konstanta PHP

Suatu konstanta adalah suatu ungkapan yang memiliki nilai tetap, artinya tidak berubah selama eksekusi program. Konstanta biasanya digunakan untuk memberikan nilai tetap pada perhitungan. [Peranginangin, 2006:80]

2.3. MySQL

Dalam membangun sebuah aplikasi, pastilah dibutuhkan suatu *Database*. Hal ini berlaku juga untuk aplikasi berbasis Web yang bersifat dinamis. Di dalam *Database* inilah sistem akan menyimpan data-data, agar lebih terstruktur dan mudah dalam penggunaan jika suatu saat diperlukan. Untuk membantu dalam manipulasi dan pemrosesan data dalam *Database* agar dapat diakses, dibutuhkan suatu sistem manajemen basis data. Pada Tugas Akhir ini, penulis menggunakan MySQL.

MySQL adalah sebuah program *Database* server yang mampu menerima dan mengirimkan datanya dengan sangat cepat, *multi user* serta menggunakan perintah standar SQL (Structured *Query* Language). [Nugroho, 2005:1]

2.3.1. Kelebihan MySQL

MySQL adalah sebuah *Database* server, dapat juga berperan sebagai client sehingga sering disebut *Database* client/server dengan kemampuan dapat berjalan di sistem operasi manapun, dengan platform Windows maupun Linux. Selain itu *Database* ini memiliki beberapa kelebihan dibanding *Database* lain, diantaranya adalah: [Nugroho, 2005:3]

1. MySQL sebagai *Database management system* (DBMS)
2. MySQL sebagai Relational *Database management system* (RDBMS)
3. MySQL adalah *software Database* yang *OpenSource*, artinya program ini bersifat bebas digunakan oleh siapa saja tanpa harus membeli dan membayar lisensi kepada pembuatnya.
4. MySQL merupakan sebuah *Database* server, jadi dengan menggunakan *Database* ini dapat dihubungkan ke media internet sehingga dapat diakses dari jauh.
5. MySQL merupakan sebuah *Database* client. Selain menjadi server yang melayani *query*, MySQL juga dapat melakukan *query* yang mengakses *Database* pada server. Jadi MySQL dapat juga berperan sebagai client.
6. MySQL mampu menerima *query* yang bertumpuk dalam satu *query*.
7. MySQL merupakan *Database* yang mampu menyimpan data berkapasitas sangat besar hingga berukuran Gigabyte sekalipun.

8. MySQL menggunakan enkripsi password. Jadi *Database* ini cukup aman karena memiliki password untuk mengaksesnya.
9. MySQL didukung oleh sebuah komponen C dan perl API, sehingga *Database* ini dapat diakses melalui sebuah program aplikasi yang berada di bawah protokol internet berupa Web. Pada umumnya aplikasi yang sering digunakan adalah PHP dan Perl.

2.3.2. SQL dalam MySQL

SQL adalah sebuah bahasa *query Database* yang terstruktur. Bahasa SQL dibuat sebagai bahasa yang dapat merelasikan beberapa tabel dalam *Database* maupun merelasikan antar *Database*. Bahasa SQL ditulis langsung dalam sebuah program *Database* sehingga seorang pengguna dapat melihat langsung *query* yang diinginkan dan melihat hasilnya. SQL dibagi menjadi dua bentuk *query*, yaitu DDL dan DML. [Nugroho, 2005:5]

2.3.2.1. DDL (Data Definition Language)

DDL adalah sebuah metode *query* SQL yang digunakan untuk mendefinisikan data pada sebuah *Database*. Beberapa *query* yang digunakan adalah: [Nugroho, 2005:5]

1. CREATE.

Yaitu *query* yang digunakan untuk melakukan pembuatan tabel dan *Database*. Misalnya akan **membuat tabel admin** dengan kolom *Username*, maka *query* yang ditulis adalah sebagai berikut:

```
CREATE TABLE admin (Username varchar(15));
```

2. DROP.

Yaitu *query* yang digunakan untuk melakukan penghapusan tabel maupun *Database*. Misalnya akan **menghapus tabel admin**, maka *query* yang ditulis adalah sebagai berikut:

```
DROP TABLE admin;
```

3. ALTER.

Yaitu *query* yang digunakan untuk melakukan perubahan struktur tabel yang telah dibuat, baik menambah kolom atau mengganti nama kolom. Misalnya

akan **mengubah tabel admin dengan menambah kolom password**, maka *query* yang ditulis adalah sebagai berikut:

```
ALTER TABLE admin ADD (password varchar(32));
```

2.3.2.2. DML (Data Manipulation Language)

DML adalah sebuah metode *query* yang dapat digunakan apabila DDL telah terjadi, sehingga fungsi dari *query* ini adalah untuk melakukan manipulasi *Database* yang telah ada atau telah dibuat sebelumnya. Adapun *query* yang termasuk di dalamnya adalah: [Nugroho, 2005:6]

1. INSERT

Yaitu *query* yang digunakan untuk melakukan pemasukan data pada tabel *Database*. Misalnya akan **memasukkan nilai pada tabel admin dengan administrator di kolom Username dan r4h4s14 di kolom password**, maka *query* yang ditulis adalah sebagai berikut:

```
INSERT INTO admin VALUES ("administrator","r4h4s14");
```

2. UPDATE

Yaitu *query* yang digunakan untuk melakukan perubahan terhadap data yang ada pada tabel. Misalnya akan **mengubah nilai pada tabel admin kolom password menjadi rahasia pada baris data yang nilai Username adalah administrator**, maka *query* yang ditulis adalah sebagai berikut:

```
UPDATE admin SET password = "rahasia" WHERE Username = "administrator";
```

3. DELETE

Yaitu *query* yang digunakan untuk melakukan penghapusan data pada tabel. Penghapusan ini dapat dilakukan secara sekaligus (seluruh isi tabel) maupun hanya beberapa baris saja. Misalnya akan **menghapus nilai pada tabel admin di baris data yang nilai Username adalah administrator**, maka *query* yang ditulis adalah sebagai berikut:

```
DELETE admin WHERE Username = "administrator";
```

2.4. AJAX

Diperkenalkan oleh Jesse James Garret dari adaptive path pada tahun 2005. Ia mendeskripsikan bagaimana mengembangkan Web yang berbeda dengan metode tradisional. Ia mempublikasikan sebuah artikel yang berjudul "AJAX : A New

Approach to Web Applications”. Pada artikelnya ini garret yakin bahwa aplikasi Web dapat menutup jurang pemisah antara Web dan aplikasi desktop.

Pengembangan Web secara tradisional berkerja secara *Synchronously*, antara aplikasi dan server, yaitu setiap melakukan klik pada *Link* atau melakukan operasi submit pada *Form*. Caranya, Browser mengirim data ke server, server merespons dan seluruh halaman akan di Refresh. Sedangkan aplikasi Web yang menggunakan AJAX bekerja secara *aSynchronously*, yang berarti mengirim dan menerima data dari *User* ke server tanpa perlu membaca kembali seluruh halaman, melainkan hanya mengambil halaman yang berubah saja.

AJAX bukanlah bahasa pemrograman baru, akan tetapi merupakan teknik baru penggunaan standar yang telah ada. Dengan AJAX dapat menjadikan Web lebih baik, cepat dan menambah unsur *User friendly* dan interaktif. AJAX berbasiskan pada javascript dan *Request* HTTP. [Sunyoto, 2007:159]

2.4.1. Pengertian AJAX

AJAX (*Asynchronous JavaScript and XML*) adalah suatu teknik pemrograman berbasis Web untuk menciptakan aplikasi Web lebih interaktif dan dinamis. Tujuannya adalah memindahkan sebagian besar interaksi pada komputer Web Server, melakukan pertukaran data dengan server di belakang layar, sehingga halaman Web tidak harus dibaca ulang secara keseluruhan setiap kali seorang pengguna melakukan perubahan. [Sunyoto, 2007:III]

2.4.2. Keistimewaan AJAX

AJAX sebagai teknik pemrograman baru memiliki beberapa keistimewaan. Keistimewaan AJAX antara lain: [Sunyoto, 2007:170]

1. Membuat *query* kepada server tanpa memuat kembali seluruh halaman.
2. Mengurai dan bekerja dengan dokumen XML dan atau JSON.
3. Data yang dikirim sedikit sehingga menghemat bandwidth dan mempercepat koneksi.
4. Proses dilakukan di belakang layar.
5. Banyak didukung oleh Browser-Browser modern yang populer.
6. Aplikasi yang dibangun semakin interaktif dan dinamis.

2.4.3. Keuntungan AJAX

Penggunaan AJAX akan mendatangkan beberapa keuntungan. Keuntungan tersebut diantaranya adalah: [Sunnyoto, 2007:170]

1. High interactivity: aplikasi AJAX menjadi lebih interaktif dibanding dengan aplikasi Web konvensional.
2. High usability: Update data tidak me-reload keseluruhan halaman melainkan hanya yang relevan.
3. High speed: aplikasi AJAX lebih cepat dibanding dengan aplikasi Web konvensional.

2.4.4. Contoh penerapan AJAX

Berikut ini akan diberikan contoh sederhana penerapan AJAX pada sebuah *Form* yang mengirim *Request* ke server. Pada contoh ini akan menampilkan sebuah *Form* yang berisi **Name** dan **Time**. Setiap kali dilakukan aksi pada **Name**, maka jam pada **Time** akan langsung ter-Update. Agar lebih mudah dipahami, maka akan penulis tuliskan dalam bentuk langkah demi langkah sebagai berikut:

1. Asumsi

Akan dibuat dua buah *File* yang terletak dalam satu direktori yaitu **time.html** dan **time.php**. *File* time.html berisi kode program html yang akan menampilkan sebuah *Form* di layar. Pada *File* time.html ini dituliskan juga sebuah fungsi **AJAXfunction** yang tugasnya mengirim *Request* dari *Form* ke server. *Form* ini berisi dua masukan, yaitu **Name** dan **Time**. Setiap kita mengetikkan sebuah karakter pada masukan Name, maka fungsi AJAXfunction akan melakukan *Request* ke server dan secara otomatis masukan Time akan ter-Update.

Sedangkan *File* **time.php** berisi kode program yang fungsinya menampilkan waktu saat ini. *File* ini akan dipanggil oleh fungsi AJAXfunction yang terdapat pada *File* time.html yang kemudian akan menampilkan nilainya pada masukan Time di *File* time.html.

2. Baris kode program pada *File time.html* [Sunyoto, 2007:179]

```

<html>
<body>

<script type="text/javascript">
function ajaxFunction(){
  var xmlhttp;
  try{
    // Firefox, Opera 8.0+, Safari
    xmlhttp=new XMLHttpRequest();
  } catch (e){
    // Internet Explorer
    try{
      xmlhttp=new ActiveXObject("Msxml2.XMLHTTP");
    } catch (e){
      try{
        xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
      } catch (e){
        alert("Your Browser does not support AJAX!");
        return false;}}
    xmlhttp.onreadystatechange=function(){
      if(xmlhttp.readyState==4){
        document.myForm.time.value=xmlhttp.responseText;}}
    xmlhttp.open("GET","time.php",true);
    xmlhttp.send(null);}
</script>

<Form name="myForm">
Name: <input type="text"
onkeyup="ajaxFunction();" name="User:name" />
Time: <input type="text" name="time" />
</Form>

</body>
</html>

```

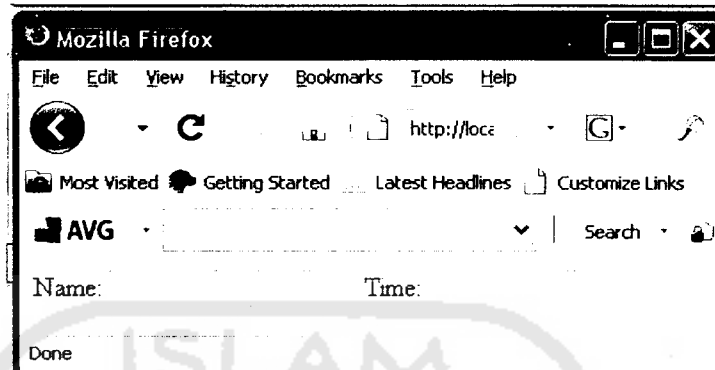
3. Baris kode program di *File time.php* [Sunyoto, 2007:180]

```

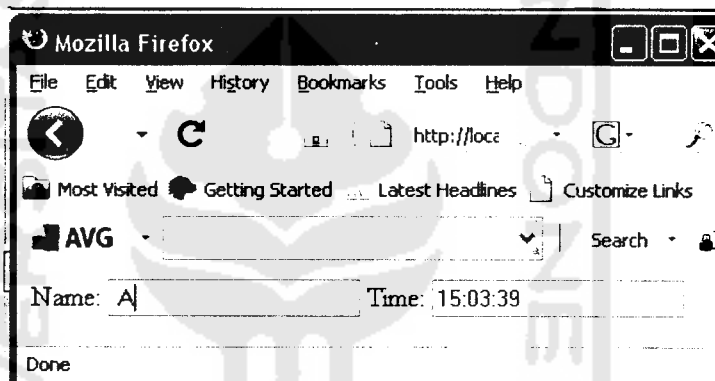
<?
echo date('H:m:s');
?>

```

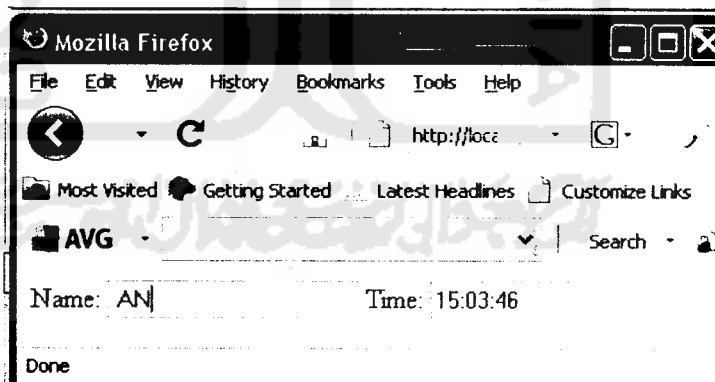
4. Keluaran



Gambar 2. 1 Tampilan awal



Gambar 2. 2 Tampilan setelah diketik satu karakter A



Gambar 2. 3 Tampilan setelah diketik dua karakter AN

5. Keterangan

Dari ketiga gambar di atas, dapat dilihat bahwa terjadi perubahan waktu pada masukan **Time** setiap kali terjadi aksi pengetikan karakter pada masukan **Name**. Sebenarnya proses yang terjadi sangatlah sederhana. Pada **time.html** dapat dilihat

baris kode program yang tercetak merah. Berikut penjelasan mengenai beberapa baris kode program tersebut.

a. `<Form name="myForm">`

Adalah sebuah *Form* dengan nama **myForm**. Pada *Form* ini terdapat dua masukan Name dan Time seperti yang terlihat di layar komputer.

b. `onkeyup="ajaxFunction();" name="Username"`

Adalah sebuah masukan yang pada layar terlihat sebagai **Name**, yang akan memanggil fungsi **ajaxFunction()** setiap kali terjadi pengetikan karakter.

c. `function ajaxFunction()`

Adalah sebuah fungsi dengan nama **ajaxFunction()**, yang mana fungsi ini akan dipanggil setiap kali ada pengetikan karakter pada masukan **Name**.

d. `xmlHttp.open("GET","time.php",true); xmlHttp.send(null);`

Adalah kode program yang terdapat dalam fungsi **ajaxFunction()** yang berfungsi melakukan pengiriman *Request* menggunakan *Method* **open()** dan **send()** ke server. Terdapat 3 parameter pada *Method* **open()**. Parameter pertama adalah *Method query* HTTP baik berupa GET, POST, HEAD atau *Method* lain yang didukung oleh server. Parameter kedua adalah letak dari halaman yang diminta. Pada parameter kedua inilah *File* time.php dipanggil. Parameter ketiga diset ketika *query* adalah *Asynchronous*. Jika diset TRUE, maka eksekusi fungsi **ajaxFunction()** akan berlanjut walau tanggapan dari server belum sampai. Ini adalah keuntungan dari AJAX. [Sunyoto, 2007:178]

e. `document.myForm.time.value=xmlHttp.responseText;`

Adalah kode program yang akan mengirimkan nilai hasil dari *File* **time.php** ke sebuah *Form* dengan nama **myForm** dan masukan dengan nama **TIME**. Pada baris kode inilah nilai dari *File* yang diminta akan dikirimkan kembali ke *File* yang melakukan *query* sehingga dapat tampil waktu saat ini pada masukan **TIME**.