

LAPORAN RESMI PRAKTIKUM TEKNOLOGI WEB

“AJAX



Nama : Rangga Rizky Ardhianto
NRP : 2110131042
Kelas : 4 D4 IT B

POLITEKNIK ELEKTRONIKA NEGERI SURABAYA
SURABAYA
2017

A. Dasar Teori

Ajax (Asynchronous JavaScript and XML) merupakan salah satu teknik berbasis JavaScript untuk membuat “background server calls” dan menerima data dari server sebatas yang diperlukan saja, tanpa harus me-load halaman secara keseluruhan. Dengan AJAX, Javascript dapat langsung berkomunikasi dengan server dengan menggunakan objek XMLHttpRequest. Dengan objek ini, javascript dapat melakukan transaksi data dengan server web, tanpa harus me-reloading halaman web tersebut secara keseluruhan.

Sebagai contoh, dalam sebuah form pengisian untuk account email baru misalnya, server bisa memberikan saran email apa yang cocok untuk user dengan nama tertentu berdasarkan kombinasi nama user tersebut tanpa harus me-load keseluruhan halaman. Contoh yang lain adalah Google Suggest, Gmail, Google Maps, dsb. Secara umum, teknologi yang menjadi dasar Ajax adalah :

- JavaScript
- Document Object Model (DOM) untuk memanipulasi bagian dari halaman HTML.
- XMLHttpRequest (XHR) Object yang memungkinkan JavaScript untuk mengakses server secara asinkron. XHR diimplementasikan secara berbeda pada browser tertentu. Sebagai contoh untuk IE6 diimplementasikan dengan : - xhr = new XMLHttpRequest(); Sedangkan untuk browser lain : - xhr = new XMLHttpRequest();

B. Percobaan

1. Membuat form pengecek nama keluarga

a. Index.html

```
1  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
2    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
3  <html xmlns="http://www.w3.org/1999/xhtml">
4  ▼ <head>
5      <title>Contoh AJAX Dasar 1</title>
6      <script type="text/javascript" src="cek.js"></script>
7  </head>
8
9  <!-- bagian untuk menampilkan form -->
10 ▼ <body onload="process()">
11     <form name="form1">
12         Siapa nama Anda <input type="text" id="namaAnda" />
13     </form>
14     <!-- bagian untuk menampilkan respon -->
15     <div id="respon" />
16 </body>
17 </html>
```

Pada file html ini mendefinisikan form untuk input dan mendefinisikan sebuah div untuk menampilkan respon. Ketika file ini diload maka akan memanggil fungsi process pada cek.js

b. Cek.js

```
var xmlhttp = createXmlHttpRequestObject();
// membuat obyek XMLHttpRequest
function createXmlHttpRequestObject()
{
    var xmlhttp;
    // cek untuk browser IE
    if(window.ActiveXObject) {
        try{
            xmlhttp = new
ActiveXObject("Microsoft.XMLHTTP");
        }
        catch (e) {
            xmlhttp = false;
        }
    }
}
```

```

// cek untuk browser Firefox atau yang lain
else{
    try{
        xmlHttp = new XMLHttpRequest();
    }
    catch (e){
        xmlHttp = false;
    }

    // muncul pesan apabila obyek XMLHttpRequest
    gagal dibuat
    if (!xmlHttp) alert("Obyek XMLHttpRequest gagal
    dibuat");
    else{
        return xmlHttp;
    }
}

// melakukan request secara asynchronous dengan
XMLHttpRequest ke
// server
function process(){
    // akan diproses hanya bila obyek XMLHttpRequest tidak
    sibuk
    if (xmlHttp.readyState == 4 || xmlHttp.readyState ==
    0){
        // mengambil nama dari text box (form)
        nama =
        encodeURIComponent(document.getElementById("namaAnda").va
        lue);
        // merequest file cek.php di server secara
        asynchronous
        xmlHttp.open("GET", "cek.php?nama=" + nama,
        true);
        // mendefinisikan metode yang dilakukan apabila
        memperoleh
        // response server
        xmlHttp.onreadystatechange =
        handleServerResponse;
        // membuat request ke server
        xmlHttp.send(null);
    }
    else{

```

```

        // Jika server sibuk, request akan dilakukan lagi
        setelah
        // satu detik
        setTimeout('process()', 1000);
    }
}

// fungsi untuk metode penanganan response dari server
function handleServerResponse(){
// jika proses request telah selesai dan menerima respon
    if (xmlHttp.readyState == 4){
        // jika request ke server sukses
        if (xmlHttp.status == 200){
            // mengambil dokumen XML yang diterima dari
            server
            xmlResponse = xmlHttp.responseXML;
            // memperoleh elemen dokumen (root elemen)
            dari xml
            xmlDocumentElement =
            xmlResponse.documentElement;
            // membaca data elemen
            hasil = xmlDocumentElement.firstChild.data;
            // akan mengupdate tampilan halaman web pada
            elemen bernama
            // respon
            document.getElementById("respon").innerHTML
            = '<i>' + hasil + '</i>';
            // request akan dilakukan lagi setelah
            // satu detik (automatic request)
            setTimeout('process()', 1000);
        }
        else{
            // akan muncul pesan apabila terjadi masalah
            dalam mengakses
            // server (selain respon 200)
            alert("Terjadi masalah dalam mengakses
            server " +
            xmlHttp.statusText);
        }
    }
}
}

```

Terdapat 3 fungsi pada file javascript ini yaitu process() , createXmlHttpRequestObject() dan handleServerResponse(). Fungsi process berisi perintah untuk melakukan http request ke cek.php dengan membawa parameter nama. Dan fungsi createXmlHttpRequestObject() digunakan untuk menginisiasi objek xml http request. Sedangkan fungsi handleServerResponse() berisi perintah untuk menghandle respons xm dari http request tersebut lalu melakukan dom pada div id respon untuk menampilkanya pada file html.

c. Cek.php

```
1 <?php
2 header('Content-Type: text/xml');
3 $nama = $_GET['nama'];
4 // membuat root tag elemen
5 echo '<response>';
6 // daftar anggota keluarga
7 $myFamily = array('ROSIHAN', 'AMALIA', 'FAZA', 'NADA');
8 // jika nama berada dalam daftar anggota keluarga
9 if (in_array(strtoupper($nama), $myFamily))
10 |   echo 'Hallo &lt;strong>'. htmlentities($nama) . '&lt;/strong>; Anda adalah anggota keluarga saya';
11 // jika nama masih kosong
12 else if (trim($nama) == "")
13 |   echo 'Hai orang asing,silakan tulis namamu';
14 // jika nama tidak ada dalam daftar anggota keluarga
15 else
16 |   echo '&lt;strong>'. htmlentities($nama) . '&lt;/strong>; Anda bukan anggota keluarga saya';
17 // menutup root tag elemen
18 echo '</response>';
19 ?>
```

File ini berisi perintah untuk menggerenate file xml dengan input parameter nama. Disini dilakukan pengecekan apakah nama yang dikirim termasuk dari array tersebut. Respon akan bergantung pada pengecekan tersebut.

2. Membuat Kalkulator Sederhana

Source Code

kalkulator.html

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Kalkulator</title>
<script type="text/javascript" src="kalkulator.js"></script>
</head>
<body>
<form name="form1">

<!-- input bilangan pertama -->
<table>
  <tr>
    <td>Angka Pertama</td>
    <td>:</td>
    <td><input type="text" id="bil1" /></td>
  </tr>
  <tr>
    <td>Angka Kedua</td>
```

```

        <td>:</td>
        <td><input type="text" id="bil2" /></td>
    </tr>
    <tr>
        <td>Operasi</td>
        <td>:</td>
        <td><select id="operasi">
            <option value="penjumlahan">+</option>
            <option value="pengurangan">-</option>
            <option value="perkalian">*</option>
            <option value="pembagian">/</option>
        </select></td>
    </tr>
</table>
<input type="button" value="Hitung" onclick="hitung()" />
</form>
<!-- bagian untuk menampilkan output -->
<p><div id="output" /></p>
</body>
</html>

```

kalkulator.js

```

var xmlHttp = createXmlHttpRequestObject();
function createXmlHttpRequestObject()
{
    var xmlHttp;
    if(window.ActiveXObject)
    {
        try
        {
            xmlHttp = new
ActiveXObject("Microsoft.XMLHTTP");
        }
        catch (e)
        {
            xmlHttp = false;
        }
    }
    else
    {
        try
        {
            xmlHttp = new XMLHttpRequest();
        }
        catch (e)
        {
            xmlHttp = false;
        }
    }

    if (!xmlHttp) alert("Obyek XMLHttpRequest gagal
dibuat");
    else
        return xmlHttp;
}

```

```

function hitung()
{
    if (xmlHttp.readyState == 4 || xmlHttp.readyState == 0)
    {
        // mengambil data input dari elemen bernama bill
        dan dikonversi
        // ke float / riil supaya dapat dioperasikan
        secara aritmatika
        bill =
parseFloat(encodeURIComponent(document.getElementById("bill")
.value));
        // mengambil data input dari elemen bernama bil2
        dan dikonversi
        // ke float / riil supaya dapat dioperasikan
        secara aritmatika
        bil2 =
parseFloat(encodeURIComponent(document.getElementById("bil2")
.value));
        // mengambil data input dari elemen bernama
        operasi
        operasi =
encodeURIComponent(document.getElementById("operasi").value);
        // proses perhitungan operasi dilakukan di script
        kalkulator.php
        xmlHttp.open("GET", "kalkulator.php?bill=" + bill
+ "&bil2=" + bil2 + "&op=" + operasi, true);
        xmlHttp.onreadystatechange =
handleServerResponse;
        xmlHttp.send(null);
    }
    else
        setTimeout('hitung()', 1000);
}
function handleServerResponse()
{
    if (xmlHttp.readyState == 4)
    {
        if (xmlHttp.status == 200)
        {
            xmlResponse = xmlHttp.responseXML;
            xmlDocumentElement =
xmlResponse.documentElement;
            hasil = xmlDocumentElement.firstChild.data;
            document.getElementById("output").innerHTML
= hasil;
            // setTimeout('process()', 1000);
        }
        else
        {
            alert("Ada masalah dalam koneksi ke server:
" + xmlHttp.statusText);
        }
    }
}
}

```


kalkulator.php

```
<?php
header('Content-Type: text/xml');
$bil1 = $_GET['bil1'];
$bil2 = $_GET['bil2'];
$op = $_GET['op'];
echo '<output>';
if ($op == "penjumlahan") $hasil = $bil1 + $bil2;
else if ($op == "pengurangan") $hasil = $bil1 - $bil2;
else if ($op == "perkalian") $hasil = $bil1 * $bil2;
else if ($op == "pembagian")
{
if ($bil2 == 0) $hasil = "(ERROR) Divide by zero";
else $hasil = $bil1 / $bil2;
}
echo "Hasil : ". $hasil;
echo '</output>';
?>
```

- Output

-

Angka Pertama :

Angka Kedua :

Operasi : * ▼

Hasil : 4

Angka Pertama :

Angka Kedua :

Operasi : + ▼

Hasil : 2

Angka Pertama :

Angka Kedua :

Operasi : - ▼

Hasil : 1