

LAPORAN HASIL PRAKTIKUM
PEMROGRAMAN WEBSITE DAN MOBILE I



NAMA : MUHAMMAD ARIFIN
NIM : 193020503025
KELAS : A
MODUL : IV

JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PALANGKA RAYA

2021

BAB I

TUJUAN DAN LANDASAN TEORI

1. Tujuan Praktikum

- 1.1.** Mahasiswa mampu membuat program dengan menggabungkan fitur PHP dan JavaScript.
- 1.2.** Mahasiswa mampu membuat program web yang dinamis.

2. Landasan Teori

AJAX digunakan untuk membuat halaman web yang cepat dan dinamis. AJAX mengizinkan halaman web diperbaharui secara asinkron oleh pertukaran data dalam jumlah kecil dengan aktifitas server dibelakang layar. Jadi, pembaharuan halaman web dilakukan tanpa merubah seluruh tampilan halaman. Halaman web yang klasik (tidak menggunakan AJAX) harus memuat seluruh halaman jika kontennya harus berubah. Contoh aplikasi penggunaan AJAX adalah seperti Google Maps, Gmail, Youtube, dan Tab Facebook. AJAX didasari oleh Standar internet dan menggunakan kombinasi dari:

- Objek XMLHttpRequest (untuk merubah data secara asinkron dengan sebuah server)
- JavaScript/DOM (untuk menampilkan atau berinteraksi dengan informasi)
- CSS (untuk memodelkan data)
- XML (sering digunakan sebagai format pengiriman data)

DOM adalah singkatan dari Document Object Model.

Kode program berikut adalah contoh penggunaan AJAX untuk membuat saran otomatis pada text input:

```

<html>
<head>
<script>
function showHint(str) {
    if (str.length == 0) {
        document.getElementById("txtHint").innerHTML =
        ""; return;
    } else {
        var xmlhttp = new XMLHttpRequest();
        xmlhttp.onreadystatechange = function() {
            if (this.readyState == 4 && this.status == 200) {
                document.getElementById("txtHint").innerHTML =

```

Gambar 1.1 HTML.

```

this.responseText;
        }
    };
    xmlhttp.open("GET", "gethint.php?q=" + str,
    true); xmlhttp.send();
}
</script>
</head>
<body>

<p><b>Mulai ketikkan sebuahnama pada field input
dibawah:</b></p> <form>
Nama depan: <input type="text"
onkeyup="showHint(this.value)"> </form>
<p>Saran: <span id="txtHint"></span></p>
</body>
</html>

```

Gambar 1.2 HTML.

```

this.responseText;
        }
    };
    xmlhttp.open("GET", "gethint.php?q=" + str,
        true); xmlhttp.send();
    }
}
</script>
</head>
<body>

<p><b>Mulai ketikkan sebuah nama pada field input
dibawah:</b></p> <form>
Nama depan: <input type="text"
onkeyup="showHint(this.value)"> </form>
<p>Saran: <span id="txtHint"></span></p>
</body>
</html>

```

Gambar 1.3 HTML.

```

<?php
    // Array with names
    $a[ ] = "Anna"; $a[
    ] = "Brittany"; $a[ ]
    = "Cinderella"; $a[ ]
    = "Diana"; $a[ ] =
    "Eva";
    $a[ ] = "Fiona";

```

Gambar 1.4 HTML.

```

$a[] = "Gunda";
$a[] = "Hege";
$a[] = "Inga";
$a[] = "Johanna";
$a[] = "Kitty";
$a[] = "Linda";
$a[] = "Nina";
$a[] = "Ophelia";
$a[] = "Petunia";
$a[] = "Amanda";
$a[] = "Raquel";
$a[] = "Cindy";
$a[] = "Doris";
$a[] = "Eve";
$a[] = "Evita";
$a[] = "Sunniva";
$a[] = "Tove";
$a[] = "Unni";
$a[] = "Violet";
$a[] = "Liza";
$a[] = "Elizabeth";
$a[] = "Ellen";
$a[] = "Wenche";
$a[] = "Vicky";

// get the q parameter from
URL $q = $_REQUEST["q"];

$hint = "";

// lookup all hints from array if $q is different from
"" if ($q != "") {
    $q = strtolower($q);
    $len=strlen($q);
    foreach($a as $name) {
        if (stristr($q, substr($name, 0, $len)))
            { if ($hint == "") {
                $hint = $name;
            } else {
                $hint .= ", $name";
            }
        }
    }
}

```

Gambar 1.5 HTML.

```

    }
}

}

// Output "no suggestion" if no hint was found or output correct
values echo $hint == "" ? "no suggestion" : $hint;
?>

```

Gambar 1.6 HTML.

AJAX dan MySQL AJAX bisa digunakan untuk komunikasi interaktif dengan sebuah database. Dibawah ini akan diberikan contoh kode program untuk mengakses database menggunakan AJAX, namun sebelum itu, perlu dibuat tabel “user” pada database “my_db” dan diisi dengan data sebagai berikut:

id	FirstName	LastName	Age	Hometown	Job
1	Peter	Griffin	41	Quahog	Brewery
2	Lois	Griffin	40	Newport	Piano Teacher
3	Joseph	Swanson	39	Quahog	Police Officer
4	Glenn	Quagmire	41	Quahog	Pilot

Gambar 1.7 Database.

Berikut adalah contoh program tersebut:

```
<html>
<head>
<script>
function showUser(str) {
    if (str == "") {
        document.getElementById("txtHint").innerHTML =
        ""; return;
    } else {
        if (window.XMLHttpRequest) {
            // code for IE7+, Firefox, Chrome, Opera,
            // Safari xmlhttp = new XMLHttpRequest();
        } else {
            // code for IE6, IE5
            xmlhttp = new ActiveXObject("Microsoft.XMLHTTP");
        }
    }
}
```

Gambar 1.8 HTML dan JAVA.

```

        xmlhttp.onreadystatechange = function() {
            if (this.readyState == 4 && this.status == 200) {
                document.getElementById("txtHint").innerHTML =
                    this.responseText; }
        };
        xmlhttp.open("GET","getuser.php?q="+str,true);
        xmlhttp.send();
    }
}
</script>
</head>
<body>

<form>
<select name="users" onchange="showUser(this.value)">
    <option value="">Select a person:</option>
    <option value="1">Peter Griffin</option> <option
value="2">Lois Griffin</option>
    <option value="3">Joseph Swanson</option>
    <option value="4">Glenn Quagmire</option>
</select>
</form>
<br>
<div id="txtHint"><b>Person info will be listed here...</b></div>

</body>
</html>

```

Gambar 1.9 HTML dan JAVA.

Penjelasan kode: Pertama, memeriksa apakah ada data orang yang terpilih. Jika tidak ada maka (str == ""), bersihkan konten dari txtHint dan keluar dari fungsi. Jika ada data orang terpilih, maka akan melakukan hal berikut:

- Membuat objek XMLHttpRequest.
- Membuat fungsi untuk dieksekusi ketika jawaban server adalah siap (ready).
- Kirim permintaan off untuk sebuah file pada server.
- Memberitahukan bahwa sebuah parameter (q) ditambahkan ke URL (dengan konten dropdown list). Contoh program diatas menggunakan file "getuser.php". File tersebut berisi kode program

untuk melakukan query pada database MySQL, Kode program dari file tersebut adalah sebagai berikut:

```
<!DOCTYPE html>
<html>
<head>
<style>
table {
    width: 100%;
    border-collapse: collapse;
}

table, td, th {
    border: 1px solid black;
    padding: 5px;
}

th {text-align: left;}
</style>
</head>
<body>

<?php
    $q = intval($_GET['q']);

    $con = mysqli_connect('localhost','peter','abc123','my_db'); if
    (!$con){
        die('Could not connect: ' . mysqli_error($con));
    }

    mysqli_select_db($con,"ajax_demo");
    $sql="SELECT * FROM user WHERE id =
    ".$q.""; $result = mysqli_query($con,$sql);

    echo "<table>
        <tr>
            <th>Firstname</th>
            <th>Lastname</th>
            <th>Age</th>
            <th>Hometown</th>
            <th>Job</th>
        </tr>";
```

Gambar 1.10 HTML dan JAVA.

```
while($row = mysqli_fetch_array($result)) {
    echo "<tr>";
    echo "<td>" . $row['FirstName'] . "</td>";
    echo "<td>" . $row['LastName'] . "</td>";
    echo "<td>" . $row['Age'] . "</td>";
    echo "<td>" . $row['Hometown'] . "</td>";
    echo "<td>" . $row['Job'] . "</td>";
    echo "</tr>";
}
echo "</table>";
mysqli_close($con);
?>
</body>
</html>
```

Gambar 1.11 HTML dan JAVA.

Penjelasan dari kode program: Ketika query dikirimkan dari JavaScript ke file PHP, hal-hal berikut terjadi:

- PHP membuka koneksi ke server MySQL.
- Menemukan data orang.
- Tabel HTML dibuat, menambahkan data ke tabel tersebut, dan mengirimkannya kembali ke placeholder “txtHint”.

AJAX dan XML AJAX bisa digunakan untuk komunikasi interaktif dengan file XML. Berikut adalah contoh program AJAX yang digunakan untuk menampilkan data XML:

```
<html>
<head>
<script>
function showCD(str) {
    if (str=="") {
        document.getElementById("txtHint").innerHTML="";
        return;
    }
    if (window.XMLHttpRequest) {
        // code for IE7+, Firefox, Chrome, Opera,
        // Safari
        xmlhttp=new XMLHttpRequest();
    } else { // code for IE6, IE5
```

Gambar 1.11 AJAX dan XML.

```

        xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
    }
    xmlhttp.onreadystatechange=function() {
        if (this.readyState==4 && this.status==200) {
            document.getElementById("txtHint").innerHTML=this.responseText;
        }
    }
    xmlhttp.open("GET","getcd.php?q="+str,true);
    xmlhttp.send();
}
</script>
</head>
<body>

<form>
Select a CD:
<select name="cds" onchange="showCD(this.value)">
<option value="">Select a CD:</option>
<option value="Bob Dylan">Bob Dylan</option>
<option value="Bee Gees">Bee Gees</option>
<option value="Cat Stevens">Cat Stevens</option>
</select>
</form>
<div id="txtHint"><b>CD info will be listed here...</b></div>

</body>
</html>

```

Gambar 1.12 AJAX dan XML.

Penjelasan kode program: Fungsi showCD() melakukan hal-hal berikut:

- Memeriksa apakah ada CD yang terpilih.
- Membuat sebuah objek XMLHttpRequest.
- Membuat fungsi yang akan dieksekusi ketika jawaban server adalah siap (ready).
- Mengirimkan permintaan off untuk sebuah file pada server.
- Memberitahukan bahwa sebuah parameter (q) ditambahkan ke URL (dengan konten dari list dropdown).

Pada kode program tersebut, terdapat file “getcd.php”. Kode program dari file tersebut memuat dokumen XML “cd_catalog.xml”, menjalankan query terhadap file XML dan mengembalikan hasilnya. File “cd_catalog.xml” bisa didownload melalui link

https://www.w3schools.com/php/cd_catalog.xml. Berikut adalah kode program dari file “getcd.php”:

```

<?php
$q=$_GET["q"];

$xmlDoc = new DOMDocument();
$xmlDoc->load("cd_catalog.xml");

$x=$xmlDoc->getElementsByTagName('ARTIST');

for ($i=0; $i<=$x->length-1; $i++) {
    //Hanya memproses element node
    if ($x->item($i)->nodeType==1){
        if ($x->item($i)->childNodes->item(0)->nodeValue == $q)
            { $y=($x->item($i)->parentNode);
            }
        }
    }

    $cd=($y->childNodes);

    for ($i=0;$i<$cd->length;$i++) {
        //Hanya memproses element node
        if ($cd->item($i)->nodeType==1){
            echo("<b>" . $cd->item($i)->nodeName . ":</b> ");
            echo($cd->item($i)->childNodes->item(0)-
            >nodeValue); echo("<br>");
        }
    }
}
?>

```

Gambar 1.13 Fungsi.

Ketika query CD dikirimkan dari JavaScript ke halaman PHP, hal-hal berikut terjadi:

- PHP membuat sebuah objek XML DOM.
- Menemukan semua element yang sesuai dengan nama yang dikirimkan dari JavaScript.
- Mengeluarkan output berupa informasi album (mengirimkan ke placeholder “txtHint”)

AJAX – Pencarian Langsung (Live Search)

AJAX bisa digunakan untuk membuat pencarian keyword yang lebih interaktif dan user-friendly. Berikut adalah contoh program untuk melakukan pencarian melalui query pada sebuah file XML “links.xml”. File ini bisa didownload pada link: <https://www.w3schools.com/php/links.xml> . Kode program dari contoh program pencarian adalah sebagai berikut:

```

<html>
<head>
<script>
function showResult(str) {
    if (str.length==0) {
        document.getElementById("livesearch").innerHTML="";
        document.getElementById("livesearch").style.border="0px";
        return;
    }
    if (window.XMLHttpRequest) {
        // code for IE7+, Firefox, Chrome, Opera,
        // Safari
        xmlhttp=new XMLHttpRequest();
    } else { // code for IE6, IE5
        xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
    }
    xmlhttp.onreadystatechange=function() {
        if (this.readyState==4 && this.status==200) {
            document.getElementById("livesearch").innerHTML=this.responseText;
            document.getElementById("livesearch").style.border="1px solid
            #A5ACB2"; }
        }
    xmlhttp.open("GET","livesearch.php?q="+str,true);
    xmlhttp.send();
}
</script>
</head>
<body>

<form>
<input type="text" size="30" onkeyup="showResult(this.value)">
<div id="livesearch"></div>
</form>

</body>
</html>

```

Gambar 1.14 AJAX.

Penjelasan kode program: Jika field input kosong (str.length==0), fungsi akan membersihkan konten dari placeholder livesearch dan keluar dari fungsi. Jika field input tidak kosong, fungsi showResult() melakukan hal-hal berikut:

- Membuat objek XMLHttpRequest.
- Membuat fungsi yang akan dieksekusi ketika jawaban server adalah siap (ready).
- Mengirimkan permintaan off untuk sebuah file pada server.
- Memberitahukan bahwa sebuah parameter (q) ditambahkan ke URL (dengan konten field input)

Contoh kode program diatas menggunakan file “livesearch.php”. Kode program dari file tersebut adalah sebagai berikut:

```
<?php
$xmlDoc=new DOMDocument();
$xmlDoc->load("links.xml");

$x=$xmlDoc->getElementsByTagName('link');

//Mengambil parameter q dari URL
$q=$_GET["q"];

//lookup semua link dari file xml jika panjang dari
q>0 if (strlen($q)>0) {
    $hint="";
    for($i=0; $i<($x->length); $i++){
        $y=$x->item($i)->getElementsByTagName('title');
        $z=$x->item($i)->getElementsByTagName('url');
        if ($y->item(0)->nodeType==1){
            //menemukan sebuah link yang sesuai dengan text yang
dicari

            if (stristr($y->item(0)->childNodes->item(0)-
                >nodeValue,$q)){ if ($hint==""){
                $hint="<a href=" .
                $z->item(0)->childNodes->item(0)->nodeValue .
                "' target='_blank'>" .
                $y->item(0)->childNodes->item(0)->nodeValue . "</a>";
```

Gambar 1.15 AJAX.

```

    }else{
        $hint=$hint . "<br /><a href="" .
        $z->item(0)->childNodes->item(0)->nodeValue .
        "" target='_blank'>" .
        $y->item(0)->childNodes->item(0)->nodeValue .
        "</a>"; }
    }
}

// Mengatur output menjadi "no suggestion" jika tidak ada petunjuk
yang ditemukan atau nilai yang sesuai
if ($hint==""){
    $response="no suggestion";
}else{
    $response=$hint;
}

//output the response
echo $response;
?>

```

Gambar 1.16 AJAX.

Penjelasan dari kode program: Jika ada text yang dikirimkan dari JavaScript ($\text{strlen}(\$q) > 0$), hal-hal berikut akan dilakukan:

- Memuat file XML menjadi objek XML DOM baru.
- Perulangan melalui semua element untuk menemukan kata yang sesuai dengan text yang dikirimkan dari JavaScript.
- Menugaskan URL yang benar dan judul dalam variabel \$response. Jika lebih dari satu yang sesuai, semua yang sesuai akan ditambahkan ke variabel tersebut.
- Jika tidak ada yang sesuai, variabel \$response akan dimasukkan nilai text “no suggestion”.

AJAX Polling

AJAX juga baik digunakan untuk menampilkan secara langsung hasil polling. Berikut adalah contoh kode program untuk melakukan polling

```

<html>
<head>
<script>
    function getVote(int) {
        if (window.XMLHttpRequest) {
            // code for IE7+, Firefox, Chrome, Opera,
            // Safari xmlhttp=new XMLHttpRequest();
        } else { // code for IE6, IE5
            xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
        }
        xmlhttp.onreadystatechange=function() {
            if (this.readyState==4 && this.status==200) {
                document.getElementById("poll").innerHTML=this.responseText;
            }
        }
        xmlhttp.open("GET","poll_vote.php?vote="+int,true);
        xmlhttp.send();
    }
</script>
</head>
<body>

<div id="poll">
<h3>Apakah kamu menyukai AJAX atau PHP?
</h3> <form>
Yes:
<input type="radio" name="vote" value="0"
onclick="getVote(this.value)"> <br>No:
<input type="radio" name="vote" value="1" onclick="getVote(this.value)">
</form>
</div>

</body>
</html>

```

Gambar 1.17 AJAX.

Fungsi getVote() melakukan hal-hal sebagai berikut:

- Membuat sebuah objek XMLHttpRequest
- Membuat fungsi untuk dieksekusi ketika jawaban server adalah siap (ready).
- Mengirimkan permintaan off untuk sebuah file pada server.
- Memberitahukan bahwa parameter (vote) ditambahkan ke URL (dengan nilai opsi yes atau no)

Kode program diatas menggunakan file “poll_vote.php”. Kode program dari file tersebut adalah sebagai berikut:

```

<?php
    $vote = $_REQUEST['vote'];

    //mendapatkan konten dari textfile
    $filename = "poll_result.txt";
    $content = file($filename);

    //meletakkan konten dalam array
    $array = explode("||", $content[0]);
    $yes = $array[0];
    $no = $array[1];

    if ($vote == 0){
        $yes = $yes + 1;
    }
    if ($vote == 1){
        $no = $no + 1;
    }

    //menyisipkan vote ke file txt
    $insertvote = $yes."||".$no;
    $fp = fopen($filename,"w");
    fputs($fp,$insertvote);
    fclose($fp);
?>

<h2>Result:</h2>
<table>
<tr>
<td>Yes:</td>
<td>

' height='20'>
    <?php echo(100*round($yes/($no+$yes),2)); ?>%

```

Gambar 1.18 AJAX.

Nilai dikirim dari JavaScript dan hal-hal berikut akan terjadi:

- Mengambil konten dari file “poll_result.txt”.
- Meletakkan konten ke dalam variabel dan menambahkan satu ke variabel yang terpilih.
- Menuliskan hasil ke file “poll_result.txt”
- Mengeluarkan representasi grafis dari hasil polling.

File text (poll_result.txt) adalah tempatn menyimpan data polling. Data tersebut disimpan dalam bentuk berikut: 0||0

Bilangan pertama merepresentasikan voting “Yes” sedangkan bilangan kedua merepresentasikan voting “No”. Jangan lupa untuk mengizinkan web browser merubah text file.

BAB II

PEMBAHASAN

Tugas Praktikum


1. Buatlah program web untuk menyimpan data survei yang disimpan di dalam database. Carilah library javascript di internet untuk menampilkan grafik, kemudian rangkum data-data hasil survei kedalam bentuk grafik.
2. Misalkan anda seorang programmer yang diminta untuk membuat program web untuk melakukan pemungutan suara pada PEMILU. Jumlah calon yang akan dipilih adalah lima orang. Buatlah program web untuk pemilihan tersebut, dimana setiap pemilih hanya bisa memilih satu kali.

Tugas 1 : Database

Dibuat 4 field yang digunakan untuk menampung beberapa data dari mahasiswa dan pelajar terkait minat atau hobi.

1. Field id digunakan untuk menampung data id dari peminat dengan tipe data int dengan maksimal 6 karakter sekaligus sebagai primary key.
2. Field nama digunakan untuk menampung data nama dari peminat dengan tipe data varchar dan maksimal 50 karakter.
3. Field profesi digunakan untuk menampung data jenis profesi dari peminat dengan tipe data varchar dan maksimal 10 karakter.
4. Field minat digunakan untuk menampung data jenis minat dari peminat dengan tipe data varchar dan maksimal 20 karakter.

Berikut adalah tampilan struktur tabel “peminat” dari database “dataminat” pada server localhost.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
<input type="checkbox"/>	1	id 	int(6)		No	None		AUTO_INCREMENT
<input type="checkbox"/>	2	nama	varchar(50)	latin1_swedish_ci	No	None		
<input type="checkbox"/>	3	profesi	varchar(10)	latin1_swedish_ci	No	None		
<input type="checkbox"/>	4	minat	varchar(20)	latin1_swedish_ci	No	None		

Gambar 2.1 Tabel peminat.

Tugas 1 : Project file

File koneksi.php

Untuk mengkoneksikan php dengan database, dibuat terlebih dulu string beberapa string yang akan menyimpan data dari server yang digunakan, username dan password dari server tersebut, serta database yang akan dikoneksikan dengan file php. file php lalu dihubungkan dengan database menggunakan sintaks *mysqli_connect* yang kemudian memanggil string yang sudah dibuat sebelumnya.

Terdapat sebuah kondisi if yang akan menampilkan sebuah pesan jika database tidak terhubung dengan file project. Berikut adalah full sintaks yang digunakan untuk mengkoneksikan php ke database.

```
koneksi.php
1  <?php
2
3  $dbhost = 'localhost';
4  $dbuser = 'root';
5  $dbpass = '';
6  $dbname = 'dataminat';
7
8  $koneksi = new mysqli($dbhost,$dbuser,$dbpass,$dbname);
9
10 if ($koneksi->connect_error)
11 {
12     die('Database Tidak Terhubung :'. $koneksi->connect_error);
13 }
14 ?>
15
```

Gambar 2.2 File koneksi.

File index.php

File ini merupakan home atau halaman pertama yang secara otomatis terbuka jika website survei diakses. Terdapat button link menuju halaman selanjutnya yaitu halaman survei yang ditampilkan melalui chart.

```
<form method="get" action="chart.php">
<input style="background-color: #ff0b2b; color: #fff;" type="submit" value="Chart" /></form>
```

Gambar 2.3 Button link ke chart.php.

Data yang ada dalam database ditampilkan ke dalam bentuk tabel pada halaman website. Untuk menampilkan data yang ada pada database, terdapat *mysql_query* yang terhubung pada tabel dalam database menggunakan query SELECT dengan perantara koneksi.

Data ditampilkan sesuai dengan tabel heading masing-masing, data ditampilkan dalam string data dan dimasukkan ke dalam tabel data menggunakan tag `<td>`. Berikut adalah sintaks yang digunakan untuk menampilkan data ke dalam bentuk tabel.

```
<table border="1" align = "left">
  <thead>
    <tr>
      <th>ID</th>
      <th>Nama</th>
      <th>Profesi</th>
      <th>Minat</th>
    </tr>
  </thead>
  <?php
    require 'koneksi.php';
    $view = $koneksi -> query("SELECT * FROM peminat");
    while($row=$view->fetch_array()){
  ?>
    <tr>
      <td><?php echo $row['id']; ?></td>
      <td><?php echo $row['nama']; ?></td>
      <td><?php echo $row['profesi']; ?></td>
      <td><?php echo $row['minat']; ?></td>
    </tr>
    <?php }?>
  </tbody>
</table>
```

Gambar 2.4 File index.

Berikut adalah sintaks yang digunakan untuk melakukan styling pada chart halaman index.

```

.row {
  display: flex;
  margin-left: -5px;
  margin-right: -5px;
}

.column {
  flex: 50%;
  padding: 5px;
}

table {
  border-collapse: collapse;
  border-spacing: 0;
  width: 50%;
  border: 1px solid #ddd;
}

th, td {
  text-align: left;
  padding: 16px;
}

tr:nth-child(even) {
  background-color: #f2f2f2;
}

```

Gambar 2.5 Styling tabel halaman index.

File chart.php

File ini merupakan halaman yang menampilkan data pada halaman sebelumnya ke dalam bentuk chart. Terdapat button link menuju halaman selanjutnya yaitu halaman survei yang ditampilkan melalui home.

```

<form method="get" action="index.php">
<input style="background-color: #ff0b2b; color: #fff;" type="submit" value="Home" /></form>

```

Gambar 2.6 Button link ke index.php.

Chart di import dari luar atau dari website dengan javascript menggunakan sintaks *src*. Berikut adalah source dari chart yang digunakan pada halaman chart.php.

```

<script src="https://code.highcharts.com/highcharts.js"></script>
<script src="https://code.highcharts.com/modules/data.js"></script>
<script src="https://code.highcharts.com/modules/exporting.js"></script>
<script src="https://code.highcharts.com/modules/accessibility.js"></script>

```

Gambar 2.7 Source chart.

Data-data yang akan ditampilkan ke dalam chart diambil melalui database dengan perantara koneksi. Terdapat query sql *"SELECT minat,COUNT(*) AS hasil FROM `peminat` GROUP BY minat"*. Query ini berfungsi untuk menampilkan jumlah dari minat masing-masing peminat. Penjumlahan dilakukan menggunakan fungsi COUNT yang hasilnya ditampilkan pada field hasil.

Selain dalam bentuk chart, data juga ditampilkan ke dalam bentuk tabel dimana terdapat 2 field yaitu minat dan jumlah peminat. Untuk field minat mengambil data dari field minat dan untuk field jumlah peminat mengambil data field hasil dari pemrosesan query sql sebelumnya. Pengambilan data dari field database menggunakan sintaks `<?php echo $row['nama_field']; ?`.

```
<figure class="highcharts-figure">
  <div id="container"></div>
  <p class="highcharts-description"> Berikut adalah hasil survei terkait minat dari beberapa orang mahasiswa dan pelajar.</p>
  <table id="datatable">
    <thead>
      <tr>
        <th>Minat</th>
        <th>Jumlah Peminat</th>
      </tr>
    </thead>
    <tbody>
      <?php
        require 'koneksi.php';
        $view = $koneksi -> query("SELECT minat,COUNT(*) AS hasil FROM `peminat` GROUP BY minat");
        while($row=$view->fetch_array()){ ?>
        <tr>
          <td><?php echo $row['minat']; ?></td>
          <td><?php echo $row['hasil']; ?></td>
        </tr>
        <?php }?>
      </tbody>
    </table>
  </figure>
```

Gambar 2.8 Chart dan tabel halaman chart.

Berikut adalah sintaks yang digunakan untuk melakukan styling pada chart halaman chart.

```

Highcharts.chart('container', {
  data: [{
    table: 'datatable'
  }],
  chart: {
    type: 'column'
  },
  title: {
    text: 'Data extracted from a HTML table in the page'
  },
  yAxis: {
    allowDecimals: false,
    title: {
      text: 'Units'
    }
  },
  tooltip: {
    formatter: function () {
      return '<b>' + this.series.name + '</b><br/>' +
        this.point.y + ' ' + this.point.name.toLowerCase();
    }
  }
});

```

Gambar 2.9 Styling chart halaman chart.

Berikut adalah sintaks yang digunakan untuk melakukan styling pada tabel halaman chart.

```

#container {
  height: 400px;
}

.highcharts-figure, .highcharts-data-table table {
  min-width: 310px;
  max-width: 800px;
  margin: 1em auto;
}

#datatable {
  font-family: Verdana, sans-serif;
  border-collapse: collapse;
  border: 1px solid #EBEBEB;
  margin: 10px auto;
  text-align: center;
  width: 100%;
  max-width: 500px;
}

#datatable caption {
  padding: 1em 0;
  font-size: 1.2em;
  color: #555;
}

#datatable th {
  font-weight: 600;
  padding: 0.5em;
}

#datatable td, #datatable th, #datatable caption {
  padding: 0.5em;
}

#datatable thead tr, #datatable tr:nth-child(even) {
  background: #f8f8f8;
}

#datatable tr:hover {
  background: #f1f7ff;
}

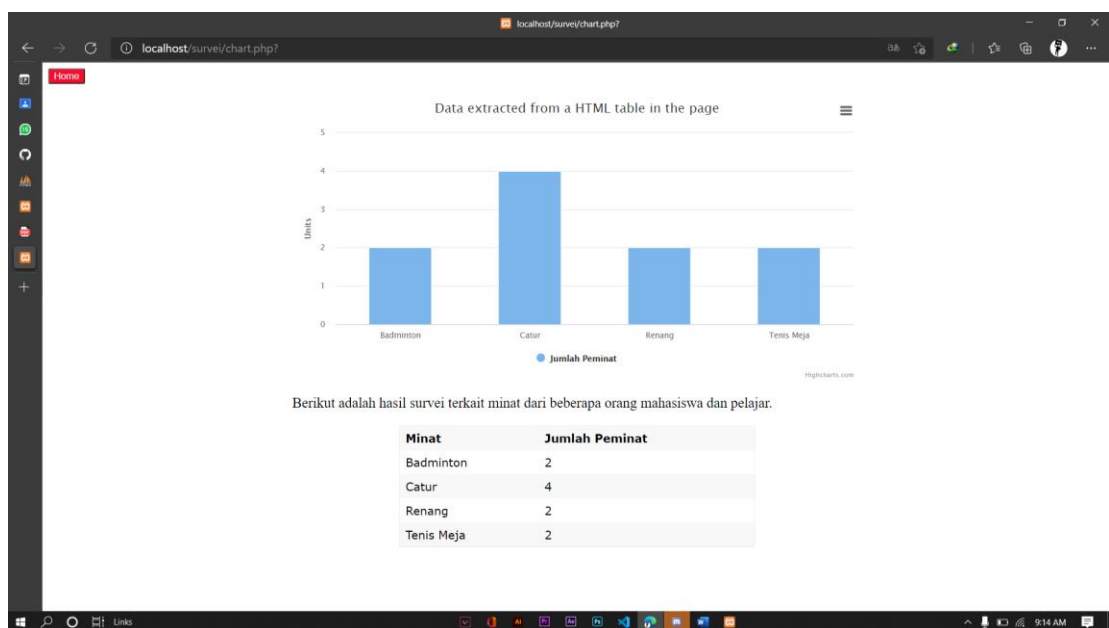
```

Gambar 2.10 Styling tabel halaman chart.

Berikut adalah tampilan halaman index dan chart setelah selesai dibuat.

ID	Nama	Profesi	Minat
1	Muhammad Arifin	Mahasiswa	Badminton
2	Arif	Pelajar	Catur
3	Ali	Pelajar	Catur
4	Andi	Mahasiswa	Badminton
5	Lia	Pelajar	Tenis Meja
6	Lina	Mahasiswa	Tenis Meja
7	Siti	Mahasiswa	Catur
8	Budi	Pelajar	Renang
9	Aldo	Pelajar	Renang
10	Gio	Pelajar	Catur

Gambar 2.11 Halaman index.




Gambar 2.12 Halaman chart.

Tugas 2 : Database

Dibuat 4 field yang digunakan untuk menampung beberapa data dari mahasiswa dan pelajar terkait minat atau hobi.

1. Field `id_suara` digunakan untuk menampung data identitas dari suara sebagai primary key yang mencegah duplikasi data voting dengan tipe data `int` dan maksimal 11 karakter.
2. Field `id_pemilih` digunakan untuk menampung data identitas dari pemilih dengan tipe data `int` dan maksimal 11 karakter.
3. Field `pilihan` digunakan untuk menampung data calon yang akan dipilih dengan tipe data `int` dan maksimal 11 karakter.

Berikut adalah tampilan struktur tabel “peminat” dari database “dataminat” pada server localhost.

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
<input type="checkbox"/>	1	id_suara 	int(11)			No	None		AUTO_INCREMENT
<input type="checkbox"/>	2	id_pemilih	int(11)			No	None		
<input type="checkbox"/>	3	pilihan	int(11)			No	None		

Gambar 2.13 Tabel pemilu.

Tugas 2 : Project file

File koneksi.php

Untuk mengkoneksikan php dengan database, dibuat terlebih dulu string beberapa string yang akan menyimpan data dari server yang digunakan, username dan password dari server tersebut, serta database yang akan dikoneksikan dengan file php. file php lalu dihubungkan dengan database menggunakan sintaks `mysqli_connect` yang kemudian memanggil string yang sudah dibuat sebelumnya.

Terdapat sebuah kondisi `if` yang akan menampilkan sebuah pesan jika database tidak terhubung dengan file project. Berikut adalah full sintaks yang digunakan untuk mengkoneksikan php ke database.

```

koneksi.php
1  <?php
2      //koneksi ke database
3      $koneksi = mysqli_connect("localhost","root","","pemilu");
4
5      // Check connection
6      if (mysqli_connect_errno()){
7          echo "Koneksi database gagal : " . mysqli_connect_error();
8      }
9  ?>

```

Gambar 2.14 File koneksi.

File index.php

Berikut adalah sintaks yang digunakan untuk melakukan voting calon, dibuat kolom id_pemilih untuk melakukan input id dari pemilih, dengan lebar 20% dari tanda “:”, dan tanda “:” itu sendiri berjarak 2% dari field input. Untuk input type atau jenis input pada field ini adalah number seperti pada database yang menggunakan tipe data int.

Terdapat 5 daftar calon ketua yang dibuat ke dalam option. Dengan adanya option, calon bisa dipilih dengan klik pada nama calon tanpa harus mengetikkan nama calon itu sendiri. jenis input pada field ini adalah submit yang berarti data akan dikirim jika button diklik.

```

<header>
  <h2>Web PEMILU</h2>
</header>
<form method="post">
  <table width="75%" cellpadding="5px">
    <tr colspan="4">
      <td width="20%">id_pemilih</td>
      <td width="2%"> : </td>
      <td colspan="2"><input class="lebar" type="number" name="id_pemilih"/></td>
    </tr>
    <tr colspan="4">
      <td width="20%">Calon Ketua</td>
      <td width="2%"> : </td>
      <td colspan="2"><select name="pilihan" class="lebar">
        <option value="Dio">1. Dio</option>
        <option value="JoJo">2. JoJo</option>
        <option value="Brando">3. Brando</option>
        <option value="Anto">4. Anto</option>
        <option value="Leo">5. Leo</option>
      </select>
      <td>
        <button class="btn btn-primary mb-10" name="kirim" type="submit" >Kirim</button>
      </td>
    </tr>
  </table>
</form>

```

Gambar 2.15 Field input.

Data voting yang sudah di input akan ditampilkan pada tabel yang dibuat menggunakan sintaks seperti di bawah. Tabel memiliki *border* sebesar 1px dengan *align* di posisi kiri atau *left*.

Data ditampilkan ke dalam tabel menggunakan query sql *"SELECT pilihan,COUNT(*) AS hasil FROM `suara` GROUP BY pilihan"* dengan perantara koneksi. Query ini berfungsi untuk menampilkan jumlah dari total suara yang didapat masing-masing calon. Penjumlahan dilakukan menggunakan fungsi COUNT yang hasilnya ditampilkan pada field hasil.

Terdapat 2 field yaitu calon dan total. Untuk field calon mengambil data dari field pilihan dan untuk field total mengambil data field hasil dari pemrosesan query sql sebelumnya. Pengambilan data dari field database menggunakan sintaks `<?php echo $row['nama_field']; ?`.

```
<table id="datatable" border="1" align = "left">
  <thead>
    <tr>
      <th>Calon</th>
      <th>Total</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td><?php
        require 'koneksi.php';
        $view = $koneksi -> query("SELECT pilihan,COUNT(*) as Hasil FROM suara GROUP by pilihan");
        while($row=$view->fetch_array()){
          ?>
          <td><?= $row['pilihan']; ?></td>
          <td><?= $row['Hasil']; ?></td>
        </tr>
      <?php }?>
    </tbody>
  </table>
```

Gambar 2.16 Tabel output.

Pada bagian ini, button kirim yang ada pada field input memiliki metode `$_POST`. Data yang sudah di input pada field input membuat sebuah antrian data yang kemudian ditangkap menggunakan `$_POST` dan dikirim ke database.

Data yang di input akan diperiksa menggunakan kondisi *if* yang jika `$data` bernilai kurang dari atau sama dengan 0 terpenuhi, maka input akan dilakukan

karena data yang akan di input adalah data baru lalu dijalankan query sql *INSERT* ke dalam field-field yang ada pada database dan akan muncul alert atau pemberitahuan bahwa “*Data berhasil ditambah*”.

Jika kondisi *\$data* bernilai lebih dari atau sama dengan 0, maka input tidak bisa dilakukan karena sudah ada data yang memiliki primary key serupa dengan input yang akan dikirim. Alert akan muncul untuk memberitahukan bahwa id sudah digunakan.

```
<?php
require './koneksi.php';
if (isset($_POST["kirim"])) {

    $pilihan = $_POST['pilihan'];
    $id_pemilih = $_POST['id_pemilih'];

    $data1 = mysqli_query($koneksi,"select * from suara where id_pemilih='$id_pemilih'");

    $cek = mysqli_num_rows($data1);

    if($cek <= 0){

        $data =mysqli_query($koneksi,"INSERT INTO `suara` (`id_suara`, `id_pemilih`, `pilihan`) VALUES ('', '$id_pemilih', '$pilihan')");

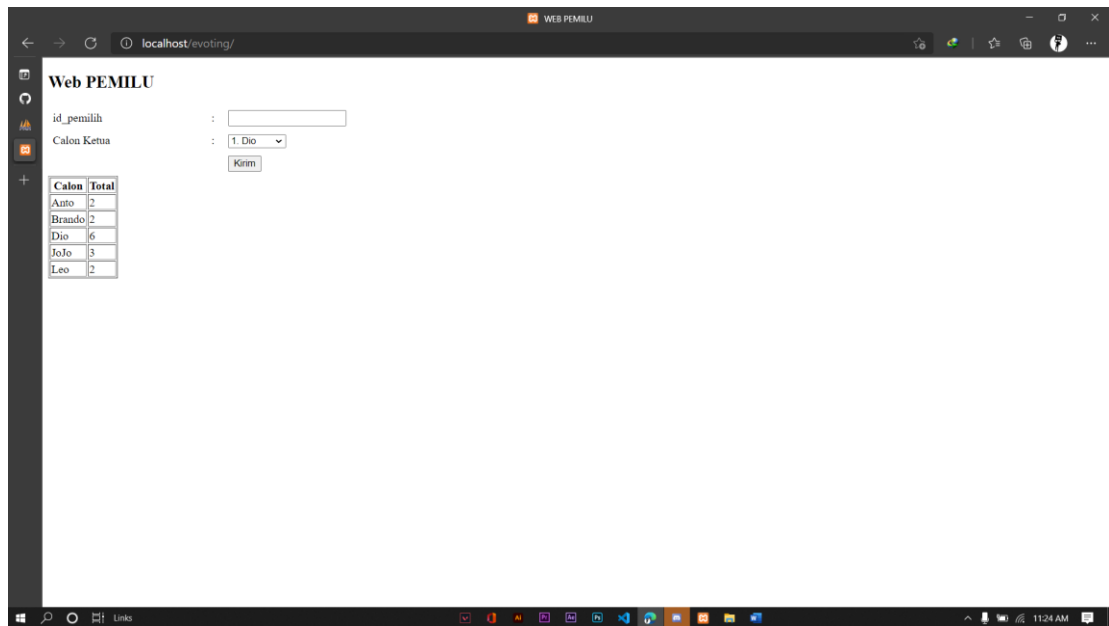
        if ($data) {
            ?>
            <script language="javascript">
            alert("Data Berhasil Ditambah");
            </script>
        <?php

    }}else if ($cek >= 0){
        ?>
        <script language="javascript">
        alert("Maaf Id sudah digunakan ");
        </script>
    <?php

}
}
?>
</body>
</html>
```

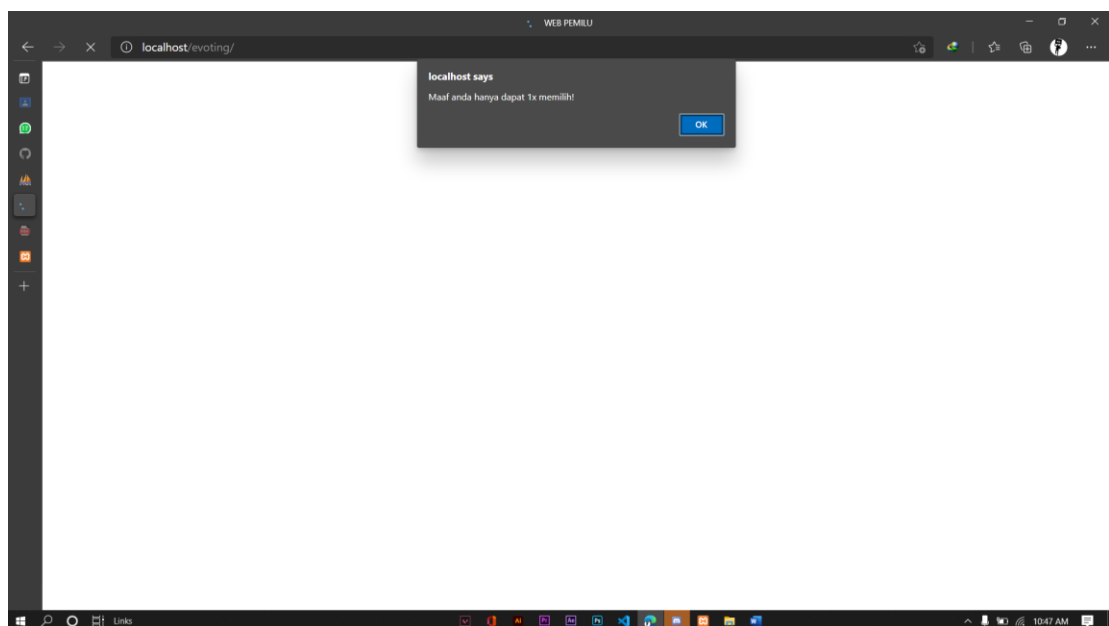
Gambar 2.17 Metode input, kondisi, dan alert.

Berikut adalah tampilan halaman website setelah selesai dibuat.



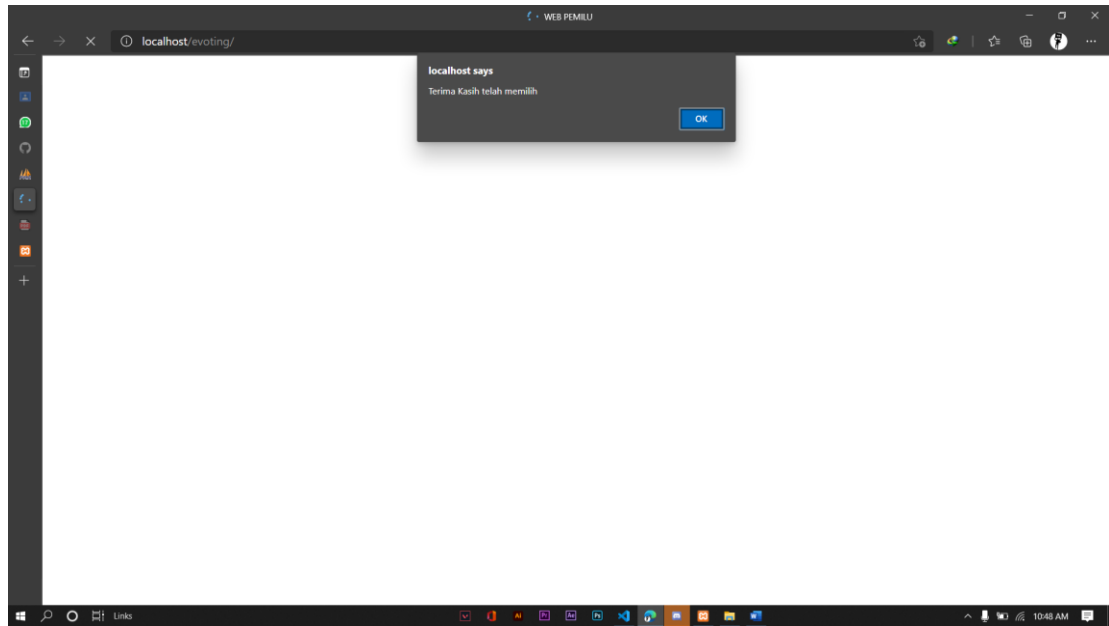
Gambar 2.18 Halaman pemilihan.

Berikut adalah kondisi jika terjadi duplikasi id_suara.



Gambar 2.19 Input tidak bisa dilakukan.

Berikut adalah kondisi jika voting berhasil dilakukan.



Gambar 2.20 Input bisa dilakukan.

BAB III
KESIMPULAN

DAFTAR PUSTAKA

Praktikum, K. (n.d.). *MODUL PRAKTIKUM PEMROGRAMAN WEB I Jurusan Teknik Informatika Fakultas Teknik Universitas Palangka Raya.*

LAMPIRAN

```
<html>
<head>
<script>
function showHint(str) {
    if (str.length == 0) {
        document.getElementById("txtHint").innerHTML =
            ""; return;
    } else {
        var xmlhttp = new XMLHttpRequest();
        xmlhttp.onreadystatechange = function() {
            if (this.readyState == 4 && this.status == 200) {
                document.getElementById("txtHint").innerHTML =
```

Gambar 1.1 HTML.

```
this.responseText;
        }
    };
    xmlhttp.open("GET", "gethint.php?q=" + str,
        true); xmlhttp.send();
}
</script>
</head>
<body>

<p><b>Mulai ketikkan sebuah nama pada field input
dibawah:</b></p> <form>
Nama depan: <input type="text"
onkeyup="showHint(this.value)"> </form>
<p>Saran: <span id="txtHint"></span></p>
</body>
</html>
```

Gambar 1.2 HTML.

```

this.responseText;
    }
    };
    xmlhttp.open("GET", "gethint.php?q=" + str,
    true); xmlhttp.send();
}
}
</script>
</head>
<body>

<p><b>Mulai ketikkan sebuah nama pada field input
dibawah:</b></p> <form>
Nama depan: <input type="text"
onkeyup="showHint(this.value)"> </form>
<p>Saran: <span id="txtHint"></span></p>
</body>
</html>

```

Gambar 1.3 HTML.

```

<?php
    // Array with names
    $a[ ] = "Anna"; $a[
    ] = "Brittany"; $a[ ]
    = "Cinderella"; $a[ ]
    = "Diana"; $a[ ] =
    "Eva";
    $a[ ] = "Fiona";

```

Gambar 1.4 HTML.

```

$a[] = "Gunda";
$a[] = "Hege";
$a[] = "Inga";
$a[] = "Johanna";
$a[] = "Kitty";
$a[] = "Linda";
$a[] = "Nina";
$a[] = "Ophelia";
$a[] = "Petunia";
$a[] = "Amanda";
$a[] = "Raquel";
$a[] = "Cindy";
$a[] = "Doris";
$a[] = "Eve";
$a[] = "Evita";
$a[] = "Sunniva";
$a[] = "Tove";
$a[] = "Unni";
$a[] = "Violet";
$a[] = "Liza";
$a[] = "Elizabeth";
$a[] = "Ellen";
$a[] = "Wenche";
$a[] = "Vicky";

// get the q parameter from
URL $q = $_REQUEST["q"];

$hint = "";

// lookup all hints from array if $q is different from
"" if ($q != "") {
    $q = strtolower($q);
    $len=strlen($q);
    foreach($a as $name) {
        if (strcasecmp($q, substr($name, 0, $len)))
            { if ($hint == "") {
                $hint = $name;
            } else {
                $hint .= ", $name";
            }
        }
    }
}

```

Gambar 1.5 HTML.

```

    }
}

// Output "no suggestion" if no hint was found or output correct
values echo $hint == "" ? "no suggestion" : $hint;
?>

```

Gambar 1.6 HTML.

id	FirstName	LastName	Age	Hometown	Job
1	Peter	Griffin	41	Quahog	Brewery
2	Lois	Griffin	40	Newport	Piano Teacher
3	Joseph	Swanson	39	Quahog	Police Officer
4	Glenn	Quagmire	41	Quahog	Pilot

Gambar 1.7 Database.

```

<html>
<head>
<script>
function showUser(str) {
    if (str == "") {
        document.getElementById("txtHint").innerHTML =
            ""; return;
    } else {
        if (window.XMLHttpRequest) {
            // code for IE7+, Firefox, Chrome, Opera,
            // Safari
            xmlhttp = new XMLHttpRequest();
        } else {
            // code for IE6, IE5
            xmlhttp = new ActiveXObject("Microsoft.XMLHTTP");
        }
    }
}

```

Gambar 1.8 HTML dan JAVA.

```

        xmlhttp.onreadystatechange = function() {
            if (this.readyState == 4 && this.status == 200) {
                document.getElementById("txtHint").innerHTML =
                    this.responseText; }
        };
        xmlhttp.open("GET","getuser.php?q="+str,true);
        xmlhttp.send();
    }
}
</script>
</head>
<body>

<form>
<select name="users" onchange="showUser(this.value)">
    <option value="">Select a person:</option>
    <option value="1">Peter Griffin</option> <option
value="2">Lois Griffin</option>
    <option value="3">Joseph Swanson</option>
    <option value="4">Glenn Quagmire</option>
</select>
</form>
<br>
<div id="txtHint"><b>Person info will be listed here...</b></div>

</body>
</html>

```

Gambar 1.9 HTML dan JAVA.

```

<!DOCTYPE html>
<html>
<head>
<style>
table {
    width: 100%;
    border-collapse: collapse;
}

table, td, th {
    border: 1px solid black;
    padding: 5px;
}

th {text-align: left;}
</style>
</head>
<body>

<?php
    $q = intval($_GET['q']);

    $con = mysqli_connect("localhost","peter","abc123","my_db"); if
    (!$con){
        die('Could not connect: ' . mysqli_error($con));
    }

    mysqli_select_db($con,"ajax_demo");
    $sql="SELECT * FROM user WHERE id =
    ".$q.""; $result = mysqli_query($con,$sql);

    echo "<table>
        <tr>
            <th>Firstname</th>
            <th>Lastname</th>
            <th>Age</th>
            <th>Hometown</th>
            <th>Job</th>
        </tr>";

```

Gambar 1.10 HTML dan JAVA.

```

        while($row = mysqli_fetch_array($result)) {
            echo "<tr>";
            echo "<td>" . $row['FirstName'] . "</td>";
            echo "<td>" . $row['LastName'] . "</td>";
            echo "<td>" . $row['Age'] . "</td>";
            echo "<td>" . $row['Hometown'] . "</td>";
            echo "<td>" . $row['Job'] . "</td>";
            echo "</tr>";
        }
        echo "</table>";
        mysqli_close($con);
    ?>
</body>
</html>

```

Gambar 1.11 HTML dan JAVA.

```

<html>
<head>
<script>
function showCD(str) {
    if (str=="") {
        document.getElementById("txtHint").innerHTML="";
        return;
    }
    if (window.XMLHttpRequest) {
        // code for IE7+, Firefox, Chrome, Opera,
        // Safari xmlhttp=new XMLHttpRequest();
    } else { // code for IE6, IE5

```

Gambar 1.11 AJAX dan XML.

```

        xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
    }
    xmlhttp.onreadystatechange=function() {
        if (this.readyState==4 && this.status==200) {
            document.getElementById("txtHint").innerHTML=this.responseText;
        }
    }
    xmlhttp.open("GET","getcd.php?q="+str,true);
    xmlhttp.send();
}
</script>
</head>
<body>

<form>
Select a CD:
<select name="cds" onchange="showCD(this.value)">
<option value="">Select a CD:</option>
<option value="Bob Dylan">Bob Dylan</option>
<option value="Bee Gees">Bee Gees</option>
<option value="Cat Stevens">Cat Stevens</option>
</select>
</form>
<div id="txtHint"><b>CD info will be listed here...</b></div>

</body>
</html>

```

Gambar 1.12 AJAX dan XML.


```

<?php
    $q=$_GET["q"];

    $xmlDoc = new DOMDocument();
    $xmlDoc->load("cd_catalog.xml");

    $x=$xmlDoc->getElementsByTagName('ARTIST');

    for ($i=0; $i<=$x->length-1; $i++) {
        //Hanya memproses element node
        if ($x->item($i)->nodeType==1){
            if ($x->item($i)->childNodes->item(0)->nodeValue == $q)
                { $y=($x->item($i)->parentNode);
                }
        }
    }

    $cd=($y->childNodes);

    for ($i=0;$i<$cd->length;$i++) {
        //Hanya memproses element node
        if ($cd->item($i)->nodeType==1){
            echo("<b>" . $cd->item($i)->nodeName . ":</b> ");
            echo($cd->item($i)->childNodes->item(0)-
            >nodeValue); echo("<br>");
        }
    }
?>

```

Gambar 1.13 Fungsi.

```

<html>
<head>
<script>
function showResult(str) {
    if (str.length==0) {
        document.getElementById("livesearch").innerHTML="";
        document.getElementById("livesearch").style.border="0px";
        return;
    }
    if (window.XMLHttpRequest) {
        // code for IE7+, Firefox, Chrome, Opera,
        // Safari xmlhttp=new XMLHttpRequest();
    } else { // code for IE6, IE5
        xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
    }
    xmlhttp.onreadystatechange=function() {
        if (this.readyState==4 && this.status==200) {
            document.getElementById("livesearch").innerHTML=this.responseText;
            document.getElementById("livesearch").style.border="1px solid
            #A5ACB2"; }
        }
    xmlhttp.open("GET","livesearch.php?q="+str,true);
    xmlhttp.send();
}
</script>
</head>
<body>

<form>
<input type="text" size="30" onkeyup="showResult(this.value)">
<div id="livesearch"></div>
</form>

</body>
</html>

```

Gambar 1.14 AJAX.

```

<?php
$xmlDoc=new DOMDocument();
$xmlDoc->load("links.xml");

$x=$xmlDoc->getElementsByTagName('link');

//Mengambil parameter q dari URL
$q=$_GET["q"];

//lookup semua link dari file xml jika panjang dari
q>0 if (strlen($q)>0) {
    $hint="";
    for($i=0; $i<($x->length); $i++){
        $y=$x->item($i)->getElementsByTagName('title');
        $z=$x->item($i)->getElementsByTagName('url');
        if ($y->item(0)->nodeType==1){
            //menemukan sebuah link yang sesuai dengan text yang
dicari

            if (stristr($y->item(0)->childNodes->item(0)-
                >nodeValue,$q)){ if ($hint==""){
                $hint="<a href=" .
                $z->item(0)->childNodes->item(0)->nodeValue .
                " target='_blank'>" .
                $y->item(0)->childNodes->item(0)->nodeValue . "</a>";
            }
        }
    }
}

```

Gambar 1.15 AJAX.

```

    }else{
        $hint=$hint . "<br /><a href="" .
        $z->item(0)->childNodes->item(0)->nodeValue .
        "" target='_blank'>" .
        $y->item(0)->childNodes->item(0)->nodeValue .
        "</a>"; }
    }
}

// Mengatur output menjadi "no suggestion" jika tidak ada petunjuk
yang ditemukan atau nilai yang sesuai
if ($hint==""){
    $response="no suggestion";
}else{
    $response=$hint;
}

//output the response
echo $response;
?>

```

Gambar 1.16 AJAX.

```

<html>
<head>
<script>
    function getVote(int) {
        if (window.XMLHttpRequest) {
            // code for IE7+, Firefox, Chrome, Opera,
            // Safari xmlhttp=new XMLHttpRequest();
        } else { // code for IE6, IE5
            xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
        }
        xmlhttp.onreadystatechange=function() {
            if (this.readyState==4 && this.status==200) {
                document.getElementById("poll").innerHTML=this.responseText;
            }
        }
        xmlhttp.open("GET","poll_vote.php?vote="+int,true);
        xmlhttp.send();
    }
</script>
</head>
<body>

<div id="poll">
<h3>Apakah kamu menyukai AJAX atau PHP?
</h3> <form>
Yes:
<input type="radio" name="vote" value="0"
onclick="getVote(this.value)"> <br>No:
<input type="radio" name="vote" value="1" onclick="getVote(this.value)">
</form>
</div>

</body>
</html>

```

Gambar 1.17 AJAX

```

<?php
    $vote = $_REQUEST['vote'];

    //mendapatkan konten dari textfile
    $filename = "poll_result.txt";
    $content = file($filename);

    //meletakkan konten dalam array
    $array = explode("||", $content[0]);
    $yes = $array[0];
    $no = $array[1];

    if ($vote == 0){
        $yes = $yes + 1;
    }
    if ($vote == 1){
        $no = $no + 1;
    }


    //menyisipkan vote ke file txt
    $insertvote = $yes."||".$no;
    $fp = fopen($filename,"w");
    fputs($fp,$insertvote);
    fclose($fp);
?>

<h2>Result:</h2>
<table>
<tr>
<td>Yes:</td>
<td>

' height='20'>
    <?php echo(100*round($yes/($no+$yes),2)); ?>%

```

Gambar 1.18 AJAX.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
<input type="checkbox"/> 1	id 	int(6)			No	None		AUTO_INCREMENT
<input type="checkbox"/> 2	nama	varchar(50)	latin1_swedish_ci		No	None		
<input type="checkbox"/> 3	profesi	varchar(10)	latin1_swedish_ci		No	None		
<input type="checkbox"/> 4	minat	varchar(20)	latin1_swedish_ci		No	None		

Gambar 2.1 Tabel peminat.

```

koneksi.php
1  <?php
2
3  $dbhost = 'localhost';
4  $dbuser = 'root';
5  $dbpass = '';
6  $dbname = 'dataminat';
7
8  $koneksi = new mysqli($dbhost,$dbuser,$dbpass,$dbname);
9
10 if ($koneksi->connect_error)
11 {
12     die('Database Tidak Terhubung :'. $koneksi->connect_error);
13 }
14 ?>
15

```

Gambar 2.2 File koneksi.

```

<form method="get" action="chart.php">
<input style="background-color: #ff0b2b; color: #fff;" type="submit" value="Chart" /></form>

```

Gambar 2.3 Button link ke chart.php.

```

<table border="1" align = "left">
  <thead>
    <tr>
      <th>ID</th>
      <th>Nama</th>
      <th>Profesi</th>
      <th>Minat</th>
    </tr>
  </thead>
  <?php
    require 'koneksi.php';
    $view = $koneksi -> query("SELECT * FROM peminat");
    while($row=$view->fetch_array()){
  >
    <tr>
      <td><?php echo $row['id']; ?></td>
      <td><?php echo $row['nama']; ?></td>
      <td><?php echo $row['profesi']; ?></td>
      <td><?php echo $row['minat']; ?></td>
    </tr>
    <?php }?>
  </tbody>
</table>

```

Gambar 2.4 File index.

```

.row {
  display: flex;
  margin-left: -5px;
  margin-right: -5px;
}

.column {
  flex: 50%;
  padding: 5px;
}

table {
  border-collapse: collapse;
  border-spacing: 0;
  width: 50%;
  border: 1px solid #ddd;
}

th, td {
  text-align: left;
  padding: 16px;
}

tr:nth-child(even) {
  background-color: #f2f2f2;
}

```

Gambar 2.5 Styling tabel halaman index.

```

<form method="get" action="index.php">
<input style="background-color: #ff0b2b; color: #fff;" type="submit" value="Home" /></form>

```

Gambar 2.6 Button link ke index.php.

```

<script src="https://code.highcharts.com/highcharts.js"></script>
<script src="https://code.highcharts.com/modules/data.js"></script>
<script src="https://code.highcharts.com/modules/exporting.js"></script>
<script src="https://code.highcharts.com/modules/accessibility.js"></script>

```

Gambar 2.7 Source chart.


```

<figure class="highcharts-figure">
  <div id="container"></div>
  <p class="highcharts-description"> Berikut adalah hasil survei terkait minat dari beberapa orang mahasiswa dan pelajar.</p>
  <table id="datatable">
    <thead>
      <tr>
        <th>Minat</th>
        <th>Jumlah Peminat</th>
      </tr>
    </thead>
    <tbody>
      <?php
        require 'koneksi.php';
        $view = $koneksi->query("SELECT minat,COUNT(*) AS hasil FROM `peminat` GROUP BY minat");
        while($row=$view->fetch_array()){ ?>
          <tr>
            <td><?php echo $row['minat']; ?></td>
            <td><?php echo $row['hasil']; ?></td>
          </tr>
        <?php ?>
      </tbody>
    </table>
  </figure>

```

Gambar 2.8 Chart dan tabel halaman chart.

```

Highcharts.chart('container', {
  data: [
    {
      table: 'datatable'
    }
  ],
  chart: {
    type: 'column'
  },
  title: {
    text: 'Data extracted from a HTML table in the page'
  },
  yAxis: {
    allowDecimals: false,
    title: {
      text: 'Units'
    }
  },
  tooltip: {
    formatter: function () {
      return '<b>' + this.series.name + '</b><br/>' +
        this.point.y + ' ' + this.point.name.toLowerCase();
    }
  }
});

```

Gambar 2.9 Styling chart halaman chart.

```

#container {
  height: 400px;
}

.highcharts-figure, .highcharts-data-table table {
  min-width: 310px;
  max-width: 800px;
  margin: 1em auto;
}

#datatable {
  font-family: Verdana, sans-serif;
  border-collapse: collapse;
  border: 1px solid #EBEBEB;
  margin: 10px auto;
  text-align: center;
  width: 100%;
  max-width: 500px;
}

#datatable caption {
  padding: 1em 0;
  font-size: 1.2em;
  color: #555;
}

#datatable th {
  font-weight: 600;
  padding: 0.5em;
}

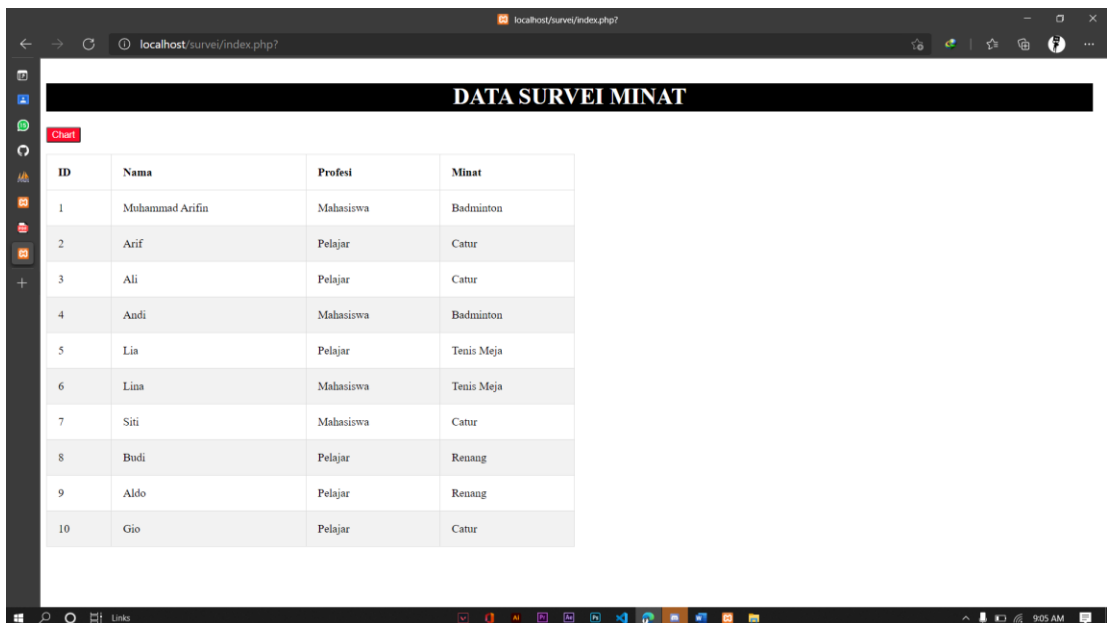
#datatable td, #datatable th, #datatable caption {
  padding: 0.5em;
}

#datatable thead tr, #datatable tr:nth-child(even) {
  background: #f8f8f8;
}

#datatable tr:hover {
  background: #f1f7ff;
}

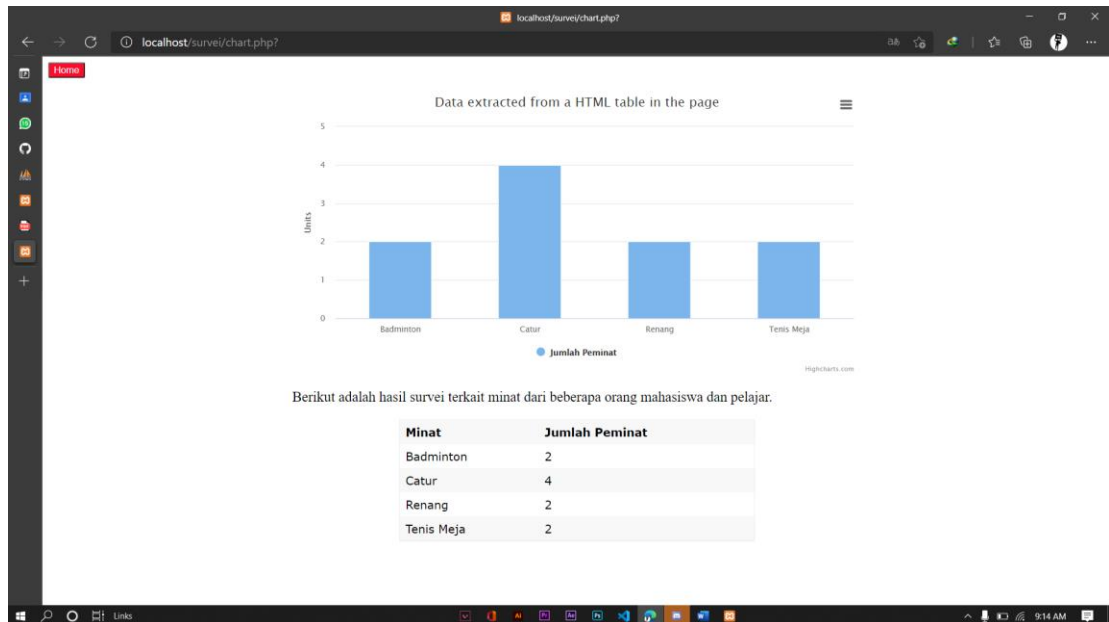
```

Gambar 2.10 Styling tabel halaman chart.



ID	Nama	Profesi	Minat
1	Muhammad Arifin	Mahasiswa	Badminton
2	Arif	Pelajar	Catur
3	Ali	Pelajar	Catur
4	Andi	Mahasiswa	Badminton
5	Lia	Pelajar	Tenis Meja
6	Lina	Mahasiswa	Tenis Meja
7	Siti	Mahasiswa	Catur
8	Budi	Pelajar	Renang
9	Aldo	Pelajar	Renang
10	Gio	Pelajar	Catur

Gambar 2.11 Halaman index.



Gambar 2.12 Halaman chart.

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
<input type="checkbox"/>	1	id_suara	int(11)			No	None		AUTO_INCREMENT
<input type="checkbox"/>	2	id_pemilih	int(11)			No	None		
<input type="checkbox"/>	3	pilihan	int(11)			No	None		

Gambar 2.13 Tabel pemilu.

```

koneksi.php
1  <?php
2      //koneksi ke database
3      $koneksi = mysqli_connect("localhost","root","","pemilu");
4
5      // Check connection
6      if (mysqli_connect_errno()){
7          echo "Koneksi database gagal : " . mysqli_connect_error();
8      }
9  ?>

```

Gambar 2.14 File koneksi.

```

<header>
  <h2>Web PEMILU</h2>
</header>
<form method="post">
  <table width="75%" cellpadding="5px">
    <tr colspan="4">
      <td width="20%">id_pemilih</td>
      <td width="2%"> : </td>
      <td colspan="2"><input class="lebar" type="number" name="id_pemilih"/></td>
    </tr>
    <tr colspan="4">
      <td width="20%">Calon Ketua</td>
      <td width="2%"> : </td>
      <td colspan="2"><select name="pilihan" class="lebar">
        <option value="Dio">1. Dio</option>
        <option value="JoJo">2. JoJo</option>
        <option value="Brando">3. Brando</option>
        <option value="Anto">4. Anto</option>
        <option value="Leo">5. Leo</option>
      </select>
    </td>
    </tr>
    <tr><td><td><td></td></td></td></tr>
    <tr><td colspan="4"><button class="btn btn-primary mb-10" name=" kirim" type="submit" >Kirim</button>
    </td></tr>
  </table>
</form>

```

Gambar 2.15 Field input.

```

<table id="datatable" border="1" align = "left">
  <thead>
    <tr>
      <th>Calon</th>
      <th>Total</th>
    </tr>
  </thead>
  <tbody>
    <?php
    require 'koneksi.php';
    $view = $koneksi -> query("SELECT pilihan,COUNT(*) as Hasil FROM suara GROUP by pilihan");
    while($row=$view->fetch_array()){
      ?>
      <tr>
        <td><? $row['pilihan']; ?></td>
        <td><? $row['Hasil']; ?></td>
      </tr>
    <?php }?>
  </tbody>
</table>

```

Gambar 2.16 Tabel output.

```

<?php
require './koneksi.php';
if (isset($_POST["kirim"])) {

    $pilihan = $_POST['pilihan'];
    $id_pemilih = $_POST['id_pemilih'];

    $data1 = mysqli_query($koneksi,"select * from suara where id_pemilih='$id_pemilih'");

    $cek = mysqli_num_rows($data1);

    if($cek <= 0){

        $data = mysqli_query($koneksi,"INSERT INTO `suara` (`id_suara`, `id_pemilih`, `pilihan`) VALUES ('', '$id_pemilih', '$pilihan')");

        if ($data) {
            ?>
            <script language="javascript">
            alert("Data Berhasil Ditambah");
            </script>
            <?php
        }}else if ($cek >= 0){
            ?>
            <script language="javascript">
            alert("Maaf Id sudah digunakan ");
            </script>
            <?php
        }
    }
    ?>
</body>
</html>

```

Gambar 2.17 Metode input, kondisi, dan alert.

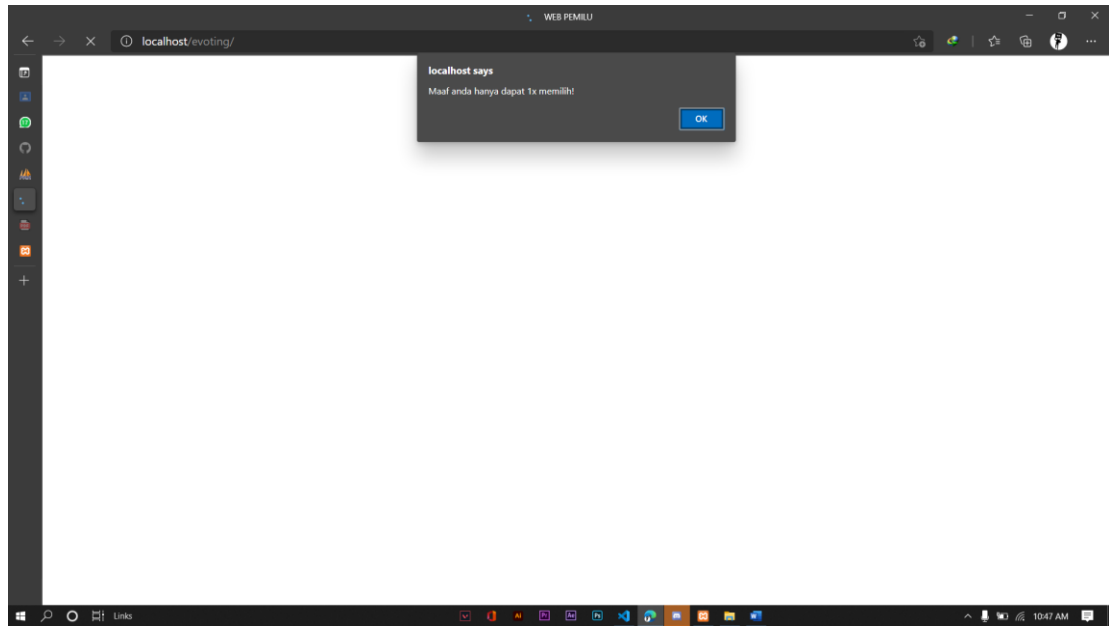
Web PEMILU

id_pemilih :

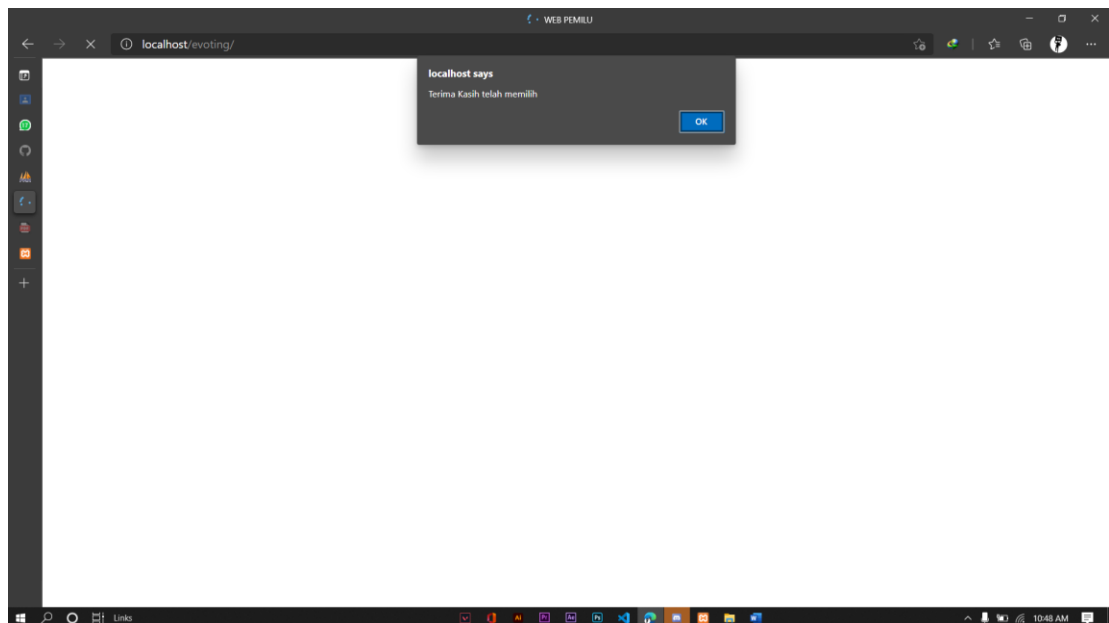
Calon Ketua :

Calon	Total
Anto	2
Brando	2
Dio	6
JoJo	3
Leo	2

Gambar 2.18 Halaman pemilihan.



Gambar 2.19 Input tidak bisa dilakukan.



Gambar 2.20 Input bisa dilakukan.