

**LAPORAN HASIL PRAKTIKUM
PEMROGRAMAN WEB DAN MOBILE I**



NAMA : KEVIN IMMANUEL RUI COSTA

NIM : 193020503030

KELAS : A

MODUL : II (FORM HANDLING)

JURUSAN/PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS PALANGKARAYA

2021

BAB I

TUJUAN DAN LANDASAN TEORI

1.1 Tujuan

1.1.1 Mahasiswa mampu membuat handling yang mampu mengolah data dari form HTML

1.1.2 Mahasiswa mampu membuat handling yang mampu mengolah data dari form HTML

1.2 Landasan Teori

PHP Form Handling merupakan hal penting dalam sebuah website, form handling digunakan untuk mengambil data dari inputan user melalui form HTML. Untuk mengakses nilai bidang formulir tertentu, bisa menggunakan variabel superglobal berikut. Variabel-variabel ini tersedia di semua cakupan pada seluruh skrip.

Superglobal	Description
<code>\$_GET</code>	Berisi daftar semua nama bidang dan nilai yang dikirim oleh formulir menggunakan metode get (mis. Melalui parameter URL).
<code>\$_POST</code>	Berisi daftar semua nama bidang dan nilai yang dikirim oleh formulir menggunakan metode posting (data tidak akan terlihat di URL).
<code>\$_REQUEST</code>	Berisi nilai-nilai dari variabel <code>\$_GET</code> dan <code>\$_POST</code> serta nilai dari variabel superglobal <code>\$_COOKIE</code> .

Form HTML dibentuk menggunakan tag, dan tag form memiliki attribute method yang digunakan untuk menentukan bagaimana cara form mengirim data ke server (data dikirim ke halaman yang diset di dalam attribute action).

`$_POST` termasuk dalam kategori Predefined Variable yang disediakan oleh library PHP. Kegunaan dari `$_POST` hampir sama dengan `$_GET` yaitu digunakan untuk mengambil suatu nilai yang dikirimkan oleh form bedanya `$_POST` mengambil nilai yang dikirim oleh form dengan `method="post"`

Semua nilai / informasi yang dikirimkan oleh form yang menggunakan method POST tidak akan terlihat oleh user yang mengakses, dikarenakan informasi yang dikirim akan tidak ditampilkan di Address Bar Web Browser. Selain \$_POST juga tidak memiliki batasan pada jumlah informasi yang dikirim.

GET dan POST membuat sebuah array (contoh array(kunci => nilai, kunci2 => nilai2, kunci3 => nilai3, ...)). Array ini menyimpan pasangan kunci/nilai, dimana kunci adalah nama-nama dari form control dan nilai adalah data input dari user. Method GET diakses menggunakan \$_GET dan method POST diakses menggunakan \$_POST. Kedua variabel ini adalah variabel superglobal, yang selalu bisa diakses, tanpa memperhatikan lingkup dan bisa diakses dari fungsi, class atau file yang berbeda tanpa harus melakukan teknik khusus. \$_GET adalah sebuah array dari variabel yang dikirimkan ke skrip melalui parameter URL. \$_POST adalah sebuah array dari variabel yang dikirimkan ke skrip melalui method HTTP POST.

Informasi dikirim dari sebuah form dengan method GET bisa dilihat oleh semua orang (semua nama dan nilai variabel ditampilkan di URL). GET juga memiliki batas pada jumlah informasi yang dikirim. Batasannya adalah sekitar 2000 karakter. Namun, karena variabel ditunjukkan di URL, ia memungkinkan untuk dilakukan bookmark halaman. Dalam beberapa kasus, hal ini sangat bermanfaat. GET bisa digunakan untuk mengirimkan data yang tidak sensitif.

Informasi yang dikirim dari sebuah form dengan method POST tidak bisa dilihat oleh siapapun (semua nama-nama atau nilai-nilai tertanam didalam body request HTTP) dan tidak memiliki batasan jumlah informasi yang akan dikirim. POST juga mendukung fungsionalitas lanjutan seperti dukungan untuk input biner multi-part ketika sedang melakukan upload file ke server. Namun, karena variabel tidak ditampilkan di URL, tidak mungkin untuk dilakukan bookmark halaman (data tidak ter-bookmark). Developer lebih baik menggunakan POST untuk mengirimkan data form.

BAB II

PEMBAHASAN

Pada praktikum ini, diminta untuk membuat program web untuk menginputkan username dan password menggunakan form dan penanganan input data dengan kriteria sebagai berikut:

1. username yang diinputkan tidak boleh lebih dari tujuh karakter.
2. password yang diinputkan harus terdiri dari huruf kapital, huruf kecil, angka dan karakter khusus.
3. Jumlah karakter password tidak boleh kurang dari sepuluh karakter.

Berikut adalah sintaks dari program web tersebut,

```
<!DOCTYPE html>
<head>
  <title>Modul 2</title>
</head>
<body>
<form method="post" name="Login_Form">
  <table>
    <td><h3>Login</h3></td>
    <tr></tr>
    <td>Username : </td>
    <td><input name="username" type="text" placeholder="Username
"></td>
    <tr></tr>
    <td>Password : </td>
    <td><input name="password" type="text" placeholder="Password
" ></td>
    <tr></tr>
    <td><input type="submit" value="Login"></td>
  </table>
</form>
</body>
</html>

<?php
  if ($_SERVER["REQUEST_METHOD"] == "POST"){
    $username = $_POST["username"];
    $password = $_POST["password"];
    $error = false;
```

```

if(strlen($username)>7){
    echo "Error : Username Lebih Dari 7" . "<br>";
    $error = true;

}if(strlen($password)<10){
    echo "Error : Password Kurang dari 10 Karakter". "<br>";
    $error = true;

}if (!preg_match("/[A-Z]/", $password) ) {
    echo "Error : Password Harus Mengandung Huruf Kapital ". "<br>";
    $error = true;

}if (!preg_match("/[a-z]/", $password)) {
    echo "Error : Password Harus Mengandung Huruf Kecil ". "<br>";
    $error = true;

}if (!preg_match("/[^a-zA-Z\d]/", $password)) {
    echo "Error : Password Harus Mengandung Karakter Spesial ". "<br>";
    $error = true;

}if (!preg_match("/[0-9]/", $password)) {
    echo "Error : Password Harus Mengandung Angka " . "<br>";
    $error = true;

}if ($error == false){
    echo "Login Berhasil";
}

}

?>

```

Pertama terdapat html, dengan isi, head dan body. Di dalam head, terdapat title Modul2 lalu setelah itu penutup head. Lalu body, di dalam body terdapat form, element yang digunakan untuk menampung macam-macam element yang berkaitan dengan sebuah form.

Lalu di dalam form, terdapat tabel, tabel tersebut digunakan untuk menampilkan wadah untuk menginputkan username, password serta menampilkan tombol untuk submit password serta username yang telah diinputkan. Membuat tempat untuk menginputkan username, password serta tombol submit menggunakan form input, berikut sintaksnya dalam program tersebut,

```
<td>Username : </td>
<td><input name="username" type="text" placeholder="Username"></td>
<tr></tr>
<td>Password : </td>
<td><input name="password" type="text" placeholder="Password" ></td>
<tr></tr>
<td><input type="submit" value="Login"></td>
```

Dapat dilihat pada sintaks di atas, yang pertama terdapat input dengan name="username" type="text" dan placeholder="Username", name="username" digunakan untuk sebagai nama, yang dapat dipanggil di php. Lalu type adalah tipe dari inputan tersebut, dalam hal ini, tipe inputannya adalah teks dan terakhir placeholder digunakan untuk tulisan dalam wadah untuk inputan.

Lalu terdapat input yang kedua, sama seperti input yang pertama, yang membedakan, input kedua digunakan untuk input password. Lalu terakhir input yang digunakan untuk submit username dan password berbeda tipe dengan inputan pertama dan kedua, tipe yang digunakan adalah submit, dan di bagian input ketiga ini, tidak ada name, karena inputan ketiga tidak di panggil di php.

Lalu setelah itu terakhir pada bagian html, terdapat penutup tabel, penutup form, penutup body dan terakhir penutup html.

Selanjutnya bagian php , pertama terdapat percabangan if dengan kondisi jika \$_SERVER["REQUEST_METHOD"] == "POST" dimana artinya jika request_method yang digunakan adalah post maka sintaks tersebut akan dijalankan.

Lalu didalam percabangan tersebut terdapat variabel dengan nama \$username, \$password dan \$error. Variabel \$username, memiliki nilai \$_POST["username"] dimana username didalam kurung siku tersebut merupakan username yang terdapat diinput name html sebelumnya. Jadi variabel tersebut akan mengambil nilai dari username yang telah disubmit pada bagian html. Lalu \$password, sama seperti \$username namun yang membedakan jika sebelumnya yang diambil adalah username yang diinput, pada \$password yang diambil adalah password yang diinputkan. Dan terakhir \$error dengan nilai false.

Lalu selanjutnya terdapat banyak percabangan, antara lain

```
if (strlen($username)>7){  
    echo "Error : Username Lebih Dari 7" . "<br>";  
    $error = true;  
}
```

Percabangan di atas digunakan untuk mengecek apakah panjang karakter dari \$username lebih besar dari 7 atau tidak, jika lebih besar maka program tersebut akan menampilkan tulisan dalam echo dan \$error akan bernilai true.

```
if (strlen($password)<10){  
    echo "Error : Password Kurang dari 10 Karakter". "<br>";  
    $error = true;  
}
```

Percabangan di atas digunakan untuk mengecek apakah panjang karakter dari \$password lebih kecil dari 10 atau tidak, jika lebih kecil maka program tersebut akan menampilkan tulisan dalam echo dan \$error akan bernilai true.

```
if (!preg_match("/[A-Z]/", $password) ) {  
    echo "Error : Password Harus Mengandung Huruf Kapital ". "<br>";  
    $error = true;  
}
```

Percabangan di atas digunakan untuk mengecek apakah \$password mengandung huruf kapital atau tidak, dapat dilihat terdapat fungsi preg_match, dimana fungsi ini digunakan untuk mengecek apakah suatu kecocokan terdapat dalam string atau tidak, dimana kecocokan yang dimaksud sesuai dengan yang

berada pada bagian “”. Dalam sintaks tersebut, kondisi dari kecocokannya adalah `"/[A-Z]/"`. Jika password tidak mengandung huruf kapital, maka program tersebut akan menampilkan tulisan dalam echo dan \$error akan bernilai true.

```
if (!preg_match("/[a-z]/", $password)) {  
    echo "Error : Password Harus Mengandung Huruf Kecil ". "<br>";  
    $error = true;  
}
```

Percabangan di atas digunakan untuk mengecek apakah \$password mengandung huruf kecil atau tidak, jika tidak maka program tersebut akan menampilkan tulisan dalam echo dan \$error akan bernilai true. Sintaks di atas juga menggunakan preg_match dengan kondisi `"/[a-z]/"`.

```
if (!preg_match("/[^a-zA-Z\d]/", $password)) {  
    echo "Error : Password Harus Mengandung Karakter Spesial ". "<br>";  
    $error = true;  
}
```

Percabangan di atas digunakan untuk mengecek apakah \$password mengandung karakter spesial atau tidak, jika tidak maka program tersebut akan menampilkan tulisan dalam echo dan \$error akan bernilai true. Sintaks di atas juga menggunakan preg_match dengan kondisi `"/[^a-zA-Z\d]/"`.

```
if (!preg_match("/[0-9]/", $password)) {  
    echo "Error : Password Harus Mengandung Angka " . "<br>";  
    $error = true;  
}
```

Percabangan di atas digunakan untuk mengecek apakah \$password mengandung angka atau tidak, jika tidak maka program tersebut akan menampilkan tulisan dalam echo dan \$error akan bernilai true. Sintaks di atas juga menggunakan preg_match dengan kondisi `"/[0-9]/"`.


```
if ($error == false){  
    echo "Login Berhasil";  
}
```

Percabangan terakhir digunakan jika login berhasil, dimana login berhasil tersebut semua kondisi yakni username yang diinputkan tidak boleh lebih dari tujuh karakter.password yang diinputkan harus terdiri dari huruf kapital, huruf kecil, angka dan karakter khusus.Jumlah karakter password tidak boleh kurang dari sepuluh karakter semuanya terpenuhi.

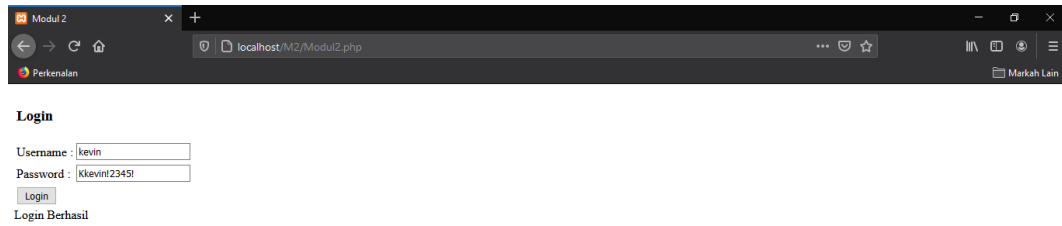
Kondisi dari percabangan tersebut adalah jika \$error == false maka echo akan ditampilkan.dapat dilihat pada percabangan sebelum-sebelumnya, terdapat \$error, dimana semua \$error bernilai true, hal tersebut dilakukan agar nilai dari variabel tersebut tidak menjadi false, karena pada percabangan-percabangan sebelumnya merupakan percabangan yang digunakan saat kondisi tidak terpenuhi atau saat kondisi error.Jika \$error diberi nilai false pada percabangan-percabangan sebelumnya, maka percabangan di atas akan ikut terpanggil meskipun kondisi dari username atau password tidak terpenuhi.

Lalu bagian akhir terdapat penutup if dan penutup php.Berikut adalah tampilan dari web dengan menggunakan kondisi-kondisi di atas.



The screenshot shows a web browser window with the address bar displaying 'localhost/M2/Modul2.php'. The page content includes a 'Login' section with two input fields: 'Username' and 'Password'. Below these fields is a 'Login' button. The browser's address bar also shows 'Modul 2' and 'Perkenalan'.

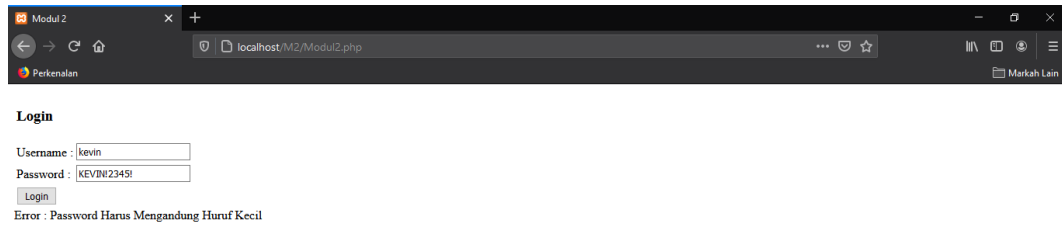
Gambar 2.1 Tampilan Awal Web Login



Gambar 2.2 Tampilan Web Login (Login Berhasil)



Gambar 2.3 Tampilan Web Login (Kurang Huruf Kapital)

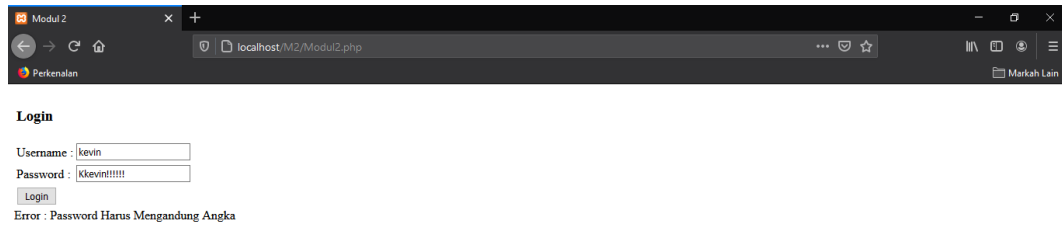


Gambar 2.4 Tampilan Web Login (Kurang Huruf Kecil)



<https://www.mozilla.org/id/firefox/central/>

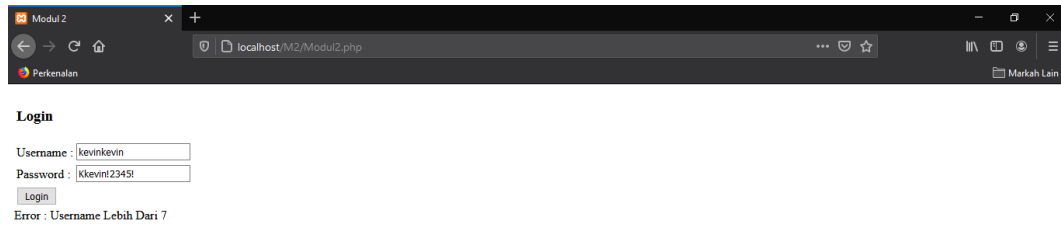
Gambar 2.5 Tampilan Web Login (Kurang Karakter Spesial)



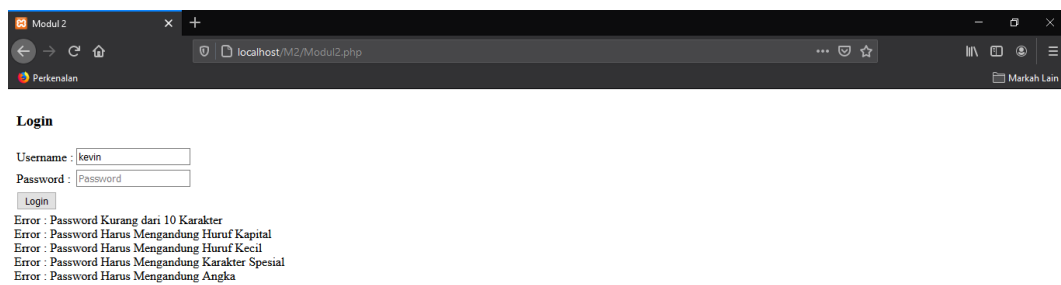
Gambar 2.6 Tampilan Web Login (Kurang Angka)



Gambar 2.7 Tampilan Web Login (Kurang dari 10 Karakter)

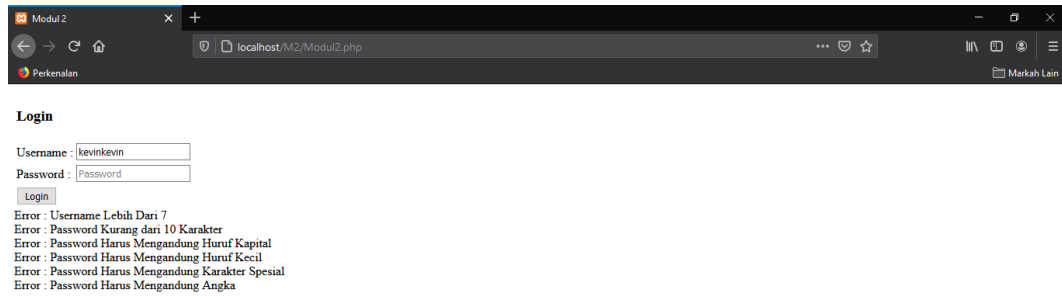


Gambar 2.8 Tampilan Web Login (Username Lebih Dari 7)



<https://www.mozilla.org/id/firefox/central/>

Gambar 2.9 Tampilan Web Login (Password Tidak Dimasukkan)



Gambar 2.10 Tampilan Web (Semua Kondisi Error)

Dapat dilihat pada gambar 2.10 dimana semua kondisi error terjadi, jika password tidak diinputkan dan jika username yang diinputkan melebihi 7 karakter.

BAB III

KESIMPULAN

Kesimpulannya, Form merupakan element dalam html yang dimana berisi kolom isian data, dalam php terdapat form handling dimana form handling digunakan untuk mengambil data dari inputan user melalui form HTML terdapat dua method untuk form handling yakni `$_POST` dan `$_GET`.

DAFTAR PUSTAKA

MODUL PRAKTIKUM PEMROGRAMAN WEB I Jurusan Teknik Informatika
Fakultas Teknik Universitas Palangka Raya, 2021

Adhitya, A. (2012). Lab-Informatika PHP Form Handling.
<https://www.lab-informatika.com/php-form-handling>

Webhozz. . Belajar PHP Form Handling - WebHozz Code.
<https://www.webhozz.com/code/php-form-handling/>

LAMPIRAN



Gambar 2.1 Tampilan Awal Web Login



Gambar 2.2 Tampilan Web Login (Login Berhasil)



Gambar 2.3 Tampilan Web Login (Kurang Huruf Kapital)



Gambar 2.4 Tampilan Web Login (Kurang Huruf Kecil)



Gambar 2.5 Tampilan Web Login (Kurang Karakter Spesial)



Gambar 2.6 Tampilan Web Login (Kurang Angka)



Gambar 2.7 Tampilan Web Login (Kurang dari 10 Karakter)



Gambar 2.8 Tampilan Web Login (Username Lebih Dari 7)



<https://www.mozilla.org/id/kefex/central/>

Gambar 2.9 Tampilan Web Login (Password Tidak Dimasukkan)



Gambar 2.10 Tampilan Web (Semua Kondisi Error)