

# **LAPORAN HASIL PRAKTIKUM PEMROGRAMAN WEB & MOBILE 1**



**NAMA : SATRIA SEPTA ARIANTO**

**NIM : 193020503026**

**KELAS : A**

**MODUL : II (FORM HANDLING)**

**JURUSAN TEKNIK INFORMATIKA  
FAKULTAS TEKNIK  
UNIVERSITAS PALANGKA RAYA  
TAHUN 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 batasan-batasan untuk menangani inputan dari form HTML.

### **1.2. Landasan Teori**

Form inputan dibuat dengan tag-tag HTML. Halaman yang mengandung form murni (tidak ada script php) tidak harus disimpan dalam bentuk php, bisa dalam bentuk html. Untuk merancang sebuah form inputan, setidaknya ada 3 (tiga) hal penting, yaitu : (Achmad Solichin. S.Kom, 2016)

#### **1. METHOD**

Method dari sebuah form menentukan bagaimana data inputan form dikirim. Method ini ada dua macam, yaitu GET dan POST. Method ini menentukan bagaimana data inputan dikirim dan diproses oleh PHP.

#### **2. ACTION**

Action dari sebuah form menentukan dimana data inputan dari form diproses. Jika action ini dikosongkan, maka dianggap proses form terjadi di halaman yang sama. Jadi halaman form dan halaman proses bisa saja dipisah atau dijadikan satu.

#### **3. SUBMIT BUTTON**

Submit button merupakan sebuah tombol (pada umumnya) yang berfungsi sebagai trigger pengiriman data dari form inputan. Jika tombol ini ditekan, maka data form akan dikirimkan (diproses) di halaman yang sudah ditentukan pada atribut action.

Variabel superglobal PHP `$_GET` dan `$_POST` digunakan untuk mengumpulkan data-form. Contoh berikut menunjukkan form HTML

sederhana dengan dua field input dan tombol submit :

```
<html>
  <body>
    <form action="welcome.php" method="post">
      Name: <input type="text" name="name"><br>
      E-mail: <input type="text" name="email"><br>
      <input type="submit">
    </form>
  </body>
</html>
```

Ketika user mengisi form, dan menekan tombol click, data form dikirim untuk memproses file PHP dengan nama “welcome.php”. Data form dikirimkan dengan method HTTP POST. Untuk menampilkan data yang sudah disubmit bisa dilakukan dengan mencetak data tersebut menggunakan perintah echo. File “welcome.php” adalah sebagai berikut :

```
<html>
  <body>
    Welcome <?php echo $_POST["name"]; ?><br>
    Your email address is: <?php echo $_POST["email"]; ?>
  </body>
</html>
```

### 1.2.1 GET vs. POST

GET dan POST membuat sebuah array (contoh array(kunci => nilai, kunci2 => nilai2, kunci3 => nilai3, ...)). Array ini menyimpan pasangan kunci/nilai, dimana kunci-kunci adalah nama-nama dari form control dan nilai-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. (Praktikum, 2021.)

### 1.2.2 Validasi Form PHP

Form HTML yang akan kita gunakan pada modul ini, mengandung bermacam-macam field input, misalnya text field yang harus diisi dan text field yang opsional, tombol pilihan (radio button), dan tombol submit. Rule atau aturan validasi untuk form diatas adalah sebagai berikut:

Field	Rule Validasi
Name	Dibutuhkan. + Harus hanya mengandung huruf dan spasi
E-mail	Dibutuhkan. + Harus mengandung sebuah alamat email yang valid dengan @ dan .
Website	Opsional. Jika ada, harus mengandung URL yang valid
Comment	Opsional. Field input multi-line (text area)
Gender	Dibutuhkan. Harus memilih salah satu

## **BAB II**

### **PEMBAHASAN**

#### **2.1 Tugas**

Buatlah 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

#### **2.2 Program Web HTML dan PHP**

Dari tugas diatas, kita diharapkan mampu membuat web untuk menginputkan username dan password menggunakan form dan penanganan input data dengan kriteria yang ada. Berikut adalah program untuk membuatnya.

Untuk program htmlnya sendiri, digunakan untuk membuat tampilannya, sedangkan untuk phpnya untuk menyimpan fungsi form dan penanganan input yang ada.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link href="https://fonts.googleapis.com/css2?family=Playfair+Display&display=swap" rel="stylesheet">
  <title>Halaman Login</title>
</head>
```

**Gambar 2.1 Head HTML**

Diatas adalah bagian kepala html untuk membuka html dan pada bagian head biasanya disertakan meta yang berhubungan dengan settingan web, link untuk mengambil sesuatu dari sumber daring maupun luring. Title untuk membuat judul web pada tab di browser.

```

<body>
  <form action="welcome.php" method="post" name="Login_Form">
    <table border="0" align="center" >
      <thead style="font-family: 'Playfair Display', sans-serif;">
        <tr>
          <td colspan="2" align="center"></td>
        </tr>
        <tr>
          <td colspan="2" align="left"><h3>Login</h3></td>
        </tr>
        <tr>
          <td align="right">Username : </td>
          <td><input name="username" type="text" placeholder="Username"></td>
        </tr>
        <tr>
          <td align="right">Password : </td>
          <td><input name="password" type="text" placeholder="Password" ></td>
        </tr>
        <tr>
          <td></td>
          <td><input name="Submit" type="submit" value="Login"></td>
        </tr>
      </thead>
    </table>
  </form>
</body>
</html>

```

**Gambar 2.2** Body HTML

Diatas adalah bagian program badan html dan bagian terpentingnya. Disinilah form handling akan dibuat, pada form terdapat action untuk menyambung ke file php lain, menggunakan method POST dan diberi nama tersendiri tergantung kegunaan form ini. Agar lebih rapi dibuatlah tabel yang diposisikan ke tengah namun tidak memiliki border sehingga tabel ini tidak akan terlihat seperti tabel. Pada bagian head tabel atau thead dideskripsikan style font yang akan digunakan.

Pada bagian awal tabel dibuat judul sebagai head yaitu login. Lalu ada 2 input yang dibuat dengan nama username dan password. Lalu ada inputan submit sebagai nilai login yang ada. Sehingga pada akhirnya file html ini ditutup.

```
<?php
    if ($_SERVER["REQUEST_METHOD"] == "POST"){
        $username = $_REQUEST["username"];
        $password = $_REQUEST["password"];
        $user = strlen($username);
        $pass = strlen($password);
        $x = false;
```

**Gambar 2.3** Form Element

Diatas adalah bagian awal file php yang berisi fungsi pemilihan if. Dalam fungsi ini ada variabel SERVER yang memiliki REQUEST\_METHOD berupa method POST. Hal ini untuk memvalidasi form nantinya jika sudah disubmit akan tervalidasi.

Variabel username dan password diinisialisasikan untuk menampung nilai username dan password nantinya. Variabel user dan pass untuk menghitung panjang string dari username dan password. Variabel x diinisiasikan sebagai nilai false.

```
if($user>7){
    echo "Username tidak boleh lebih dari 7 karakter";
    $x = true;
}
if($pass<10){
    echo "Password tidak boleh kurang dari 10 karakter";
    $x = true;
}
```

**Gambar 2.4** Penanganan Batas Input

2 fungsi if diatas adalah untuk menangani input username dan pass. Variabel yang digunakan adalah user dan pass yang berisi panjang username dan password. Sehingga bagian ini adalah untuk menangani batas input username yang tidak boleh lebih dari 7 karakter dan password yang tidak boleh kurang dari 10 karakter.

```

if (!preg_match("/[A-Z]/", $password) ) {
    echo "Password harus menyertakan huruf kapital<br> ";
    $x = true;
}
if (!preg_match("/[a-z]/", $password)) {
    echo "Password harus menyertakan huruf kecil<br>";
    $x = true;
}
if (!preg_match("/[^a-zA-Z\d]/", $password)) {
    echo "Password harus menyertakan karakter khusus<br>";
    $x = true;
}
if (!preg_match("/[0-9]/", $password)) {
    echo "Password harus menyertakan angka<br>";
    $x = true;
}

```

**Gambar 2.5** Penangan Input pada kriteria Password

Diatas adalah fungsi untuk menangani input pada bagian password. Dimana password yang diinginkan haru menyertakan huruf kapital, huruf kecil, karakter khusus dan angka.

```

        if( $x == false ){
            echo "Login Berhasil<br> ";
            echo "Selamat Datang ", $_POST["username"];
        }
    }
?>

```

**Gambar 2.6** Kriteria input terpenuhi

Diatas adalah fungsi ketika nilai variabel x tetap false karena diawal diinisiasikan false. Jika terdapat kesalahan input maka x akan menjadi true dan peringatan akan dimunculkan sedangkan jika tetap false maka login dianggap berhasil.



### **BAB III**

### **KESIMPULAN**

Penggunaan form handling dalam pembuatan form sangatlah penting, tidak hanya untuk menghindari masalah pada program, juga untuk memastikan program berjalan dengan baik dan benar. Teknik yang digunakan juga ada GET dan POST yang memiliki keunggulan di kondisi tertentu.

## DAFTAR PUSTAKA

- Achmad Solichin. S.Kom. (2016). Pemrograman Web dengan PHP dan MySQL - Achmad Solichin - Google Buku. *Universitas Budi Luhur, June*, 85.  
<https://books.google.co.id/books?id=kcD4BQAAQBAJ&printsec=frontcover&dq=aplikasi+berbasis+web+dengan&hl=id&sa=X&ved=0ahUKEwib-fT80ITYAhVBrI8KHT9GD6QQ6AEIJzAA#v=onepage&q=aplikasi+berbasis+web+dengan&f=false>
- Praktikum, K. (2021). *MODUL PRAKTIKUM PEMROGRAMAN WEB I Jurusan Teknik Informatika Fakultas Teknik Universitas Palangka Raya*.

## LAMPIRAN

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link href="https://fonts.googleapis.com/css2?family=Playfair+Display&display=swap" rel="stylesheet">
  <title>Halaman Login</title>
</head>
```

Gambar 2.1 Head HTML

```
<body>
  <form action="welcome.php" method="post" name="Login_Form">
    <table border="0" align="center" >
      <thead style="font-family: 'Playfair Display', sans-serif;">
        <tr>
          <td colspan="2" align="center"></td>
        </tr>
        <tr>
          <td colspan="2" align="left"><h3>Login</h3></td>
        </tr>
        <tr>
          <td align="right">Username : </td>
          <td><input name="username" type="text" placeholder="Username"></td>
        </tr>
        <tr>
          <td align="right">Password : </td>
          <td><input name="password" type="text" placeholder="Password" ></td>
        </tr>
        <tr>
          <td></td>
          <td><input name="Submit" type="submit" value="Login"></td>
        </tr>
      </thead>
    </table>
  </form>
</body>
</html>
```

Gambar 2.2 Body HTML

```
<?php
    if ($_SERVER["REQUEST_METHOD"] == "POST"){
        $username = $_REQUEST["username"];
        $password = $_REQUEST["password"];
        $user = strlen($username);
        $pass = strlen($password);
        $x = false;
```

Gambar 2.3 Form Element

```

if($user>7){
    echo "Username tidak boleh lebih dari 7 karakter";
    $x = true;
}
if($pass>10){
    echo "Password tidak boleh kurang dari 10 karakter";
    $x = true;
}

```

**Gambar 2.4** Penanganan Batas Input

```

if (!preg_match("/[A-Z]/", $password) ) {
    echo "Password harus menyertakan huruf kapital\n";
    $x = true;
}
if (!preg_match("/[a-z]/", $password)) {
    echo "Password harus menyertakan huruf kecil\n";
    $x = true;
}
if (!preg_match("/[^\a-zA-Z\d]/", $password)) {
    echo "Password harus menyertakan karakter khusus\n";
    $x = true;
}
if (!preg_match("/[0-9]/", $password)) {
    echo "Password harus menyertakan angka\n";
    $x = true;
}

```

**Gambar 2.5** Penangan Input pada kriteria Password

```

    if( $x == false ){
        echo "Login Berhasil, ";
        echo "selamat datang ", $_POST["username"];
    }
}
?>

```

**Gambar 2.6** Kriteria input terpenuhi