

LAPORAN HASIL PRAKTIKUM
PEMPROGRAMAN WEB DAN MOBILE I



Nama : Erlan Sebastian Usin

NIM : 193030503074

Kelas : A

Modul : II (Form Handling)

JURUSAN TEKNIK INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS PALANGKARAYA

2021

BAB I

TUJUAN DAN LANDASAN TEORI

1.1 TUJUAN

- Mahasiswa mampu membuat handling yang mampu mengolah data dari form HTML.
- Mahasiswa mampu membuat batasan-batasan untuk menangani inputan dari form HTML.

1.2 LANDASAN TEORI.

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>
```

Jika field nama diinputkan dengan Tono dan email diinputkan dengan tono@mail.com maka output yang akan tampil adalah sebagai berikut:

Welcome Budi

Your email address is tono@mail.com.

Hasil yang sama juga akan tampil dengan menggunakan method get sebagai berikut:

```
<html>

  <body>

    Welcome <?php echo $_POST["name"]; ?><br>

    Your email address is: <?php echo $_POST["email"];

  ?> </body>

</html>
```

dengan file “welcome_get.php” sebagai berikut:

```
<html>

  <body>

    Welcome <?php echo $_GET["name"];

  ?><br> Your email address is: <?php

  echo $_GET["email"];

  ?> </body>

</html>
```

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.

Validasi Form PHP

Pertimbangkan keamanan ketika memproses form PHP!

PHP Form Validation Example

* required field.

Name: *

E-mail: *

Website:

Comment:

Gender: ☐ Female ☐ Male *

Gambar 1.1 validasi 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

Kode HTML untuk membentuk Form tersebut adalah sebagai berikut:

Text Field

Field nama, email dan website adalah elemen-elemen text input, dan field komentar adalah textarea yaitu sebagai berikut:

Name: <input
type="text"
name="name"> E-
mail: <input
type="text"
name="email">

Website: <input type="text" name="website">

Comment: <textarea name="comment" rows="5" cols="40"></textarea>

Radio Button

Field jenis kelamin adalah radio button yaitu sebagai berikut:

Gender:

```
<input type="radio" name="gender" value="female">Female
```

```
<input type="radio" name="gender" value="male">Male
```

Form Element

Kode HTML untuk membentuk form pada gambar diatas adalah sebagai berikut:

```
<form method="post" action="<?php echo  
htmlspecialchars($_SERVER["PHP_SELF"]);? >">
```

Ketika form disubmit, data pada form dikirim dengan method “post”.

`$_SERVER["PHP_SELF"]` adalah variabel super global yang mengembalikan nama file dari skrip yang sedang dieksekusi. Sehingga kode form diatas mengirim data pada form ke halaman itu sendiri. Sedangkan fungsi `htmlspecialchars()` adalah fungsi yang mengkonversikan karakter-karakter spesial ke entitas HTML. Sebagai contoh, fungsi tersebut akan mengkonversikan karakter `<` dan `>` menjadi `<` dan `>`. Fungsi ini mencegah injeksi yang bisa dilakukan dengan HTML atau javascript (Cross-site Scripting Attack) pada form tersebut.

Suatu aplikasi banyak menerima masukan/input data dari pengguna, misalkan ketika registrasi kesuatu website atau ketika login pada halaman dan sebagainya. Untuk memungkinkan hal tersebut diperlukan suatu antarmuka pada klien dalam bentuk form HTML dan suatu mekanisme untuk penanganan data yang dikirim oleh pengguna pada sisi server. Form handling ialah suatu mekanisme untuk menangani suatu masukan dari form yang dikirim oleh pengguna. Form handling berhubungan dengan metode yang dikirim dan bagaimana menangani pengiriman data berdasarkan metode yang digunakan.

proses validasi data form dengan PHP, Yang menginputkan username, password, dan jumlah karakter password. Jika sintaks memasukkan digit angka di mobile numbernya tidak valid maka ada pesan error atau jika valid maka akan berhasil. Data yang dimasukkan melalui sebuah form rawan terhadap penyalahgunaan yang mengancam keamanan aplikasi web. Maka biasanya sebuah form dilengkapi dengan proses memeriksa setiap isian form agar tidak disalahgunakan untuk memasukkan data-data yang mengancam keamanan aplikasi. Selain untuk masalah keamanan, pemeriksaan form atau validasi form juga untuk memastikan bahwa semua data telah diisi dan dimasukkan sesuai ketentuan. Validasi form biasanya dilakukan atau dieksekusi pada sisi server. Validasi akan dilakukan setelah data form diisi lalu tombol submit ditekan pada komputer klien. Jika satu atau beberapa data form masih mengandung kesalahan, maka server akan mengembalikan semua data dan meminta

komputer klien untuk mengirimkan kembali data yang benar. Proses ini memakan waktu dan membutuhkan sumber daya server.

Javascript menyediakan menyediakan sebuah cara untuk memvalidasi data form pada sisi komputer kalian sebelum dikirim ke server. Dengan demikian, waktu yang diperlukan lebih sedikit begitu juga dengan penggunaan sumber daya server.

BAB II

PEMBAHASAN

2.1 PEMBAHASAN

Pada kali ini kita akan membuat program web untuk menginputkan username dan password. Berikut ini penjelasan nya:

Pertama, menunjukkan form HTML sederhana dengan tiga field input dan tombol submit:

```
html>
<body>
  <form action="" method="post">
    <input type="text" name="p" values="" placeholder="Username">
    <br>
    <input type="text" name="cp" values="" placeholder="Password">
    <br>
    <input type="text" name="s" values="" placeholder="character password 0-9">
    <br>

    <input type="submit" name="s" value="Simple validation">
  </form>
```

Pada tabel diatas merupakan langkah ketika user mengisi form, dan mengklik tombol, data form dikirimkan dengan method HTTP POST. Input type ini yaitu perintah untuk menginputkan Username, Password, dan Jumlah karakter password tidak boleh dari 10. Dan setelah menginputkan Username, password, dan character passwordnya maka langsung di submit setelah itu validasi form PHP nya.

Kedua, memvalidasi data form dengan PHP berikut ini tabelnya:

```
<?php
if(isset($_POST['s']))
{
    $n = $_POST['nm'];
    $p = $_POST['p'];
    $cp = $_POST['cp'];
    if(empty($n)) //check required
    {
        echo "please Enter Username";
    }
    if(!preg_match ("/^[a-zA-Z\s]+$/", $n)) // name is only character
    {
        echo "please Enter Name As only String";
    }
    if(!preg_match('/^[0-9]*$/', $m)) //enter only number validation
    {
        echo "Only Mobile Number ";
    }
    //check length <10 digit then error >10 digit then error
    if(strlen($m)<10)
    {
        echo "please Enter proper Mobile Number Here";
    }
    if(strlen($m)>10)
    {
        echo "Your Mobile Number is too Large Please Enter Valid...";
    }
}
```

```
//password checking both password are matched or not
if($p != $cp) //both pass are not matched then
{
    echo "Both Password are not Matched";
}
```

Pada tabel diatas merupakan proses validasi data form dengan PHP, Yang menginputkan username, password, dan jumlah karakter password. Jika sintaks memasukkan digit angka di mobile numbernya tidak valid maka ada pesan error atau jika valid maka akan berhasil. Data yang dimasukkan melalui sebuah form rawan terhadap penyalahgunaan yang mengancam keamanan aplikasi web. Maka biasanya sebuah form dilengkapi dengan proses memeriksa setiap isian form agar tidak disalahgunakan untuk memasukkan data-data yang mengancam keamanan aplikasi. Selain untuk masalah keamanan, pemeriksaan form atau validasi form juga untuk memastikan bahwa semua data telah diisi dan dimasukkan sesuai ketentuan. Validasi form biasanya dilakukan atau dieksekusi pada sisi server. Validasi akan dilakukan setelah data form diisi lalu tombol submit ditekan pada komputer klien. Jika satu atau beberapa data form masih mengandung kesalahan, maka server akan mengembalikan semua data dan meminta

komputer klien untuk mengirimkan kembali data yang benar. Proses ini memakan waktu dan membutuhkan sumber daya server. Javascript menyediakan menyediakan sebuah cara untuk memvalidasi data form pada sisi komputer kalian sebelum dikirim ke server. Dengan demikian, waktu yang diperlukan lebih sedikit begitu juga dengan penggunaan sumber daya server.

Validasi form pada umumnya melakukan dua fungsi:

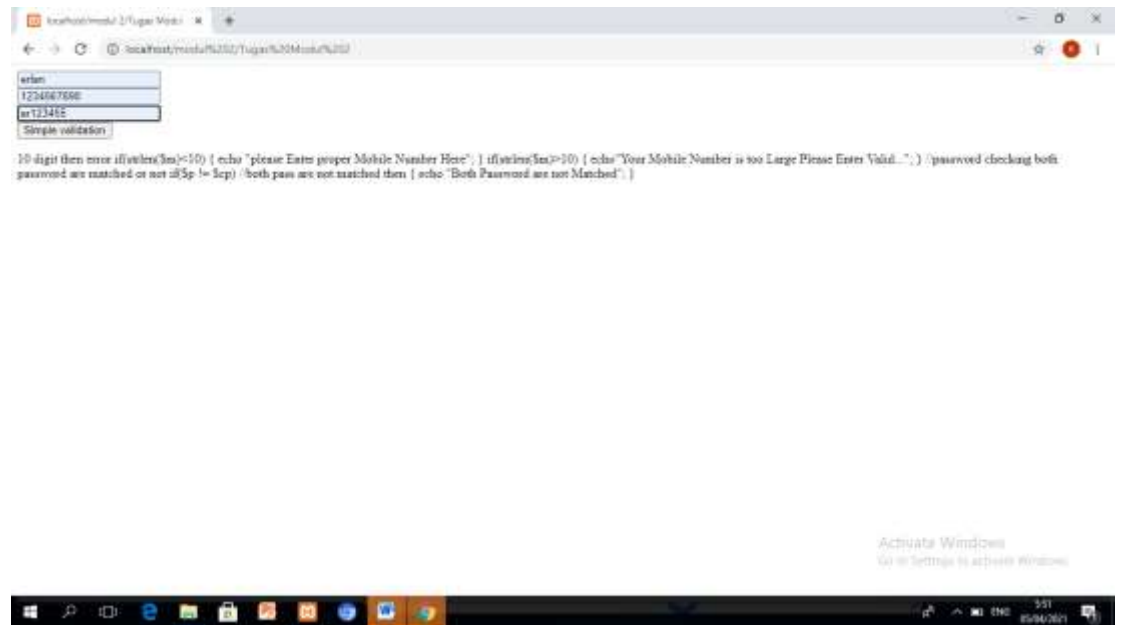
- Validasi dasar - Memeriksa semua isian form apakah semua isian sudah terisi data. Proses ini akan memeriksa satu persatu setiap isian form dan memeriksa datanya.
- Validasi format data - Memeriksa kebenaran format dan nilai data yang dimasukkan. Untuk proses pemeriksaan ini akan membutuhkan perintah javascript yang lebih kompleks.

Ketiga, mencoba program web menginputkan username dan password menggunakan form berikut ini gambarnya:



Gambar 2.1 form web.

Gambar diatas merupakan form web yang sebelum username, password , dan character password belum dimasukkan. Berikut ini gambar ketika menginputkan username, password, dan character password:



Gambar 2.2 menginputkan username, password dan lainnya.

Gambar diatas merupakan proses penginputan username, password, dan karakter password jika terdapat tulisan bawahnya error maka proses peinputannya tidak valid, begitu pula jika proses peinputannya berhasil maka valid.

BAB III

KESIMPULAN

PHP Form Handling merupakan hal penting dalam sebuah website, form handling digunakan untuk mengambil data dari inputan user melalui form HTML.

biasanya sebuah form dilengkapi dengan proses memeriksa setiap isian form agar tidak disalahgunakan untuk memasukkan data-data yang mengancam keamanan aplikasi. Selain untuk masalah keamanan, pemeriksaan form atau validasi form juga untuk memastikan bahwa semua data telah diisi dan dimasukkan sesuai ketentuan.

Validasi form biasanya dilakukan atau dieksekusi pada sisi server. Validasi akan dilakukan setelah data form diisi lalu tombol submit ditekan pada komputer klien.

Jika satu atau beberapa data form masih mengandung kesalahan, maka server akan mengembalikan semua data dan meminta

komputer kalian untuk mengirimkan kembali data yang benar. Proses ini memakan waktu dan membutuhkan sumber daya server.

Javascript menyediakan menyediakan sebuah cara untuk memvalidasi data form pada sisi komputer kalian sebelum dikirim ke server. Dengan demikian, waktu yang diperlukan lebih sedikit begitu juga dengan penggunaan sumber daya server.

Validasi form pada umumnya melakukan dua fungsi:

- Validasi dasar - Memeriksa semua isian form apakah semua isian sudah terisi data. Proses ini akan memeriksa satu persatu setiap isian form dan memeriksa datanya.
- Validasi format data - Memeriksa kebenaran format dan nilai data yang dimasukkan. Untuk proses pemeriksaan ini akan membutuhkan perintah javascript yang lebih kompleks.

DAFTAR PUSTAKA

Modul Praktikum Pemrograman Web-I Universitas Palangka Raya Teknik Informatika 2021.

Edition, F. (n.d.). *Beginning PHP and MySQL*.

Nixon, R. (2018). *Learning PHP, MySQL & JavaScript : with jQuery, CSS & HTML5*.

LAMPIRAN

