

POLITEKNIK NEGERI BALI	Job Sheet 6 FORM VALIDATION	Dosen Pengampu : Ni Ketut Pradani Gayatri Sarja, S.Kom., M.Kom.
Bidang Studi : Teknologi Rekayasa Perangkat Lunak Jurusan : Teknik Elektro		Waktu : 4 x 50 Menit
Kelas/Sem : IV/IV Th Ajaran : 2021/2022		Pertemuan 7 dan 8

A. Standar Kompetensi

Mahasiswa diharapkan mampu menerjemahkan permasalahan dalam dunia nyata ke dalam algoritma dan mengimplementasikannya ke dalam Framework CI setelah mempelajari mata kuliah ini.

B. Kompetensi Dasar

Setelah mempelajari materi ini, mahasiswa diharapkan mampu memahami dan menggunakan form validation pada framework CI.

C. Indikator

1. Mahasiswa mampu memahami penggunaan form validation
2. Mahasiswa mampu menerapkan form validation pada class view dalam framework CI

D. Peralatan

1. Seperangkat computer/laptop yang sudah terinstal Xampp, Framework CI dan aplikasi pengolah text
2. Buku Ajar
3. Job sheet

E. Teori

1. Load Library Form Validation

```
$this->load->library('form_validation');
```

2. Menentukan Rules Untuk Validasi

```
$rules = [  
    [  
        'field' => 'email',  
        'label' => 'Email Address',  
        'rules' =>  
            'required|valid_email|max_length[64]'  
    ]  
];  
$this->form_validation->set_rules($rules);
```

3. Tabel Rules Yang Disediakan Codeigniter

Rule	Parameter	Keterangan
required	Tidak	Input wajib diisi atau tidak boleh kosong
matches	Ya	Input harus match dengan parameter yang ditentukan. Contoh: matches[form_item]
regex_match	Ya	Input harus match dengan pola regex yang ditentukan. Contoh: regex_match[/regex/]
differs	Ya	Input harus berbeda dari parameter yang ditentukan. Contoh: differs[form_item]
is_unique	Ya	Input harus unik berdasarkan nama field pada database. Rule ini membutuhkan Query Builder agar bisa digunakan. Contoh: is_unique[table.field]
min_length	Ya	Panjang input harus setidaknya minimal dari nilai yang ditentukan. Contoh: min_length[3]
max_length	Ya	Panjang input harus lebih pendek dari nilai maksimal yang ditentukan. Contoh: max_length[12]
exact_length	Ya	Panjang input harus sama dengan nilai yang ditentukan. Contoh: exact_length[8]
greater_than	Ya	Input harus lebih besar dari angka yang ditentukan dan harus berupa angka. Contoh: greater_than[8]
greater_than_equal_to	Ya	Input harus lebih besar atau sama dengan nilai yang ditentukan dan harus berupa angka. Contoh: greater_than_equal_to[8]
less_than	Ya	Input harus lebih kecil dari nilai yang ditentukan dan harus berupa angka. Contoh: less_than[8]
less_than_equal_to	Ya	Input harus lebih kecil atau sama dengan dari nilai yang ditentukan dan harus berupa angka, Contoh: less_than_equal_to[8]
in_list	Ya	Input harus mengandung item seperti yang ada di dalam list, Contoh: in_list[red,blue,green]
alpha	Tidak	Input harus berupa huruf alphabet

alpha_numeric	Tidak	Input harus berupa karakter alpha-numeric
alpha_numeric_spaces	Tidak	Input harus berupa karakter alpha-numeric dan juga boleh mengandung spasi.
alpha_dash	Tidak	Input harus berupa alpabet boleh mengandung - (dash) dan _ (underscore)
numeric	Tidak	Input harus berupa karakter angka
integer	Tidak	Input harus berupa angka
decimal	Tidak	Input harus berupa angka desimal
is_natural	Tidak	Input harus berupa angka alami.
is_natural_no_zero	Tidak	Input harus berupa angka alami yang bukan nol.
valid_url	Tidak	Input harus berupa URL yang benar
valid_email	Tidak	Input harus berupa alamat email
valid_emails	Tidak	input list yang dipisah dengan koma harus berupa email yang benar
valid_ip	Tidak	Input harus berupa IP Address. Bisa dengan format 'ipv4' atau 'ipv6'.
valid_base64	Tidak	Input harus mengandung Base64

4. Menjalankan Validasi

```
$this->form_validation->run();
```

F. Keselamatan Kerja

1. Ikuti langkah kerja yang ada pada job sheet
2. Pastikan lingkungan kerja (meja dan kursi) bersih, baik sebelum maupun sesudah penggunaan
3. Siapkan alat tulis yang lengkap dan siap digunakan
4. Jika ada hal yang kurang jelas, tanyakan kepada dosen pengampu

G. Langkah Kerja

Praktek Form Validation

Membuat sebuah form input data user dengan menggunakan library Form Validation

1. Load library form validation

```
$autoload['libraries'] = array('form_validation');
```

2. Membuat model rules

```
<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class M_Form extends CI_Model {

    public $variable;

    public function __construct()
    {
        parent::__construct();
    }

    public function validation()
    {
        return [
            [
                'field' => 'nama',
                'label' => 'Nama',
                'rules' => 'required|max_length[32]'
            ],
            [
                'field' => 'email',
                'label' => 'Email',
                'rules' => 'required|valid_email|max_length[32]'
            ],
            [
                'field' => 'alamat',
                'label' => 'Alamat',
                'rules' => 'required'
            ]
        ];
    }
}
```

3. Membuat fungsi pada controller untuk menampilkan view form

```
public function index()
{
    $this->load->view('form');
}
```

4. Membuat form

```
<!DOCTYPE html>
<html>
<head>
  <title>Form</title>
</head>
<body>
  <form method="post" action="<?php echo base_url('form/input') ?>">
    <table>
      <tr>
        <td colspan="3"><?php echo form_error('nama') ?></td>
      </tr>
      <tr>
        <td><label>Nama</label></td>
        <td></td>
        <td><input type="text" name="nama" placeholder="Input Nama"></td>
      </tr>
      <tr>
        <td colspan="3"><?php echo form_error('email') ?></td>
      </tr>
      <tr>
        <td><label>Email</label></td>
        <td></td>
        <td><input type="text" name="email" placeholder="Input Email"></td>
      </tr>
      <tr>
        <td colspan="3"><?php echo form_error('alamat') ?></td>
      </tr>
      <tr>
        <td><label>Alamat</label></td>
        <td></td>
        <td><input type="text" name="alamat" placeholder="Input Alamat"></td>
      </tr>
      <tr>
        <td colspan="3"><input type="submit" value="Submit"></td>
      </tr>
    </table>
  </form>
</body>
</html>
```

5. Membuat fungsi pada controller untuk memproses validasi dan menginput data user serta menampilkannya

```
public function input()
{
  $rules = $this->M_Form->validation();
  $this->form_validation->set_rules($rules);

  if ($this->form_validation->run() == FALSE) {
    $this->load->view('form');
  }else{
    $data = array(
      'nama' => $this->input->post('nama'),
      'email' => $this->input->post('email'),
      'alamat' => $this->input->post('alamat'),
    );
    $this->load->view('hasil',$data);
  }
}
```

6. Membuat tampilan untuk menampilkan hasil input

```
<!DOCTYPE html>
<html>
<head>
  <title>Hasil</title>
</head>
<body>
  Nama : <?php echo $nama ?><br>
  Email : <?php echo $email ?><br>
  Alamat : <?php echo $alamat ?><br>
  <a href="<?php echo base_url('form') ?>">Go Back</a>
</body>
</html>
```

H. Soal/Latihan Praktek

Membuat sebuah form input dan edit barang dengan menggunakan library Form Validation kemudian simpan data ke dalam database.