

LAPORAN PRAKTIKUM
PEMROGRAMAN WEB LANJUT

FORM VALIDASI LARAVEL



NAMA : SILVI INDAH NOVITA SARI
NIM : 1841720006
KELAS : 2B – TI

D-IV TEKNIK INFORMATIKA
TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG

2020

1	Buka project Laravel “laravel-crud”
2	<p>Buka file laravel-crud/routes/web.php. buatlah 2 route, yaitu route get. dengan url ‘mahasiswa/tambah’ yang akan mengakses method tambah pada controller MahasiswaController. dan route yang 1 nya lagi kita buat route post. dengan url ‘simpan’ yang akan memproses method atau function simpan pada controller MahasiswaController</p> <pre> 22 Route::get('/mahasiswa/tambah', 'MahasiswaController@tambah'); 23 24 Route::post('/mahasiswa/simpan', 'MahasiswaController@simpan'); 25 </pre>
3	<p>Buat 2 buah method pada laravelcrud/app/Http/controller/MahasiswaController.php.</p> <pre> 23 public function tambah() 24 { 25 //memanggil view tambah 26 return view('tambah'); 27 } 28 public function simpan(Request \$request) 29 { 30 \$this->validate(\$request, [31 'nama' => 'required', 32 'nim' => 'required numeric', 33 'email' => 'required email', 34 'jurusan' => 'required' 35]); 36 //insert data ke table mahasiswa 37 DB::table('mahasiswa')->insert([38 'nama' => \$request->namamhs, 39 'nim' => \$request->nimmhs, 40 'email' => \$request->emailmhs, 41 'jurusan' => \$request->jurusanmhs 42]); 43 return view('simpan', ['data' => \$request]); 44 } 45 </pre> <p>Pada method tambah() kita membuat sebuah view yang menampilkan form penginputan sederhana. dimana pada form tersebut akan kita arahkan pemproses nya pada method simpan() menggunakan method post. seperti yang telah kita definisikan pada route.</p> <p>pada method ini kita melakukan validasi form dengan fungsi validate(). Pertama kita tangkap data-data yang dikirim ke method ini ke dalam variabel \$request. kemudian kita isi variable \$request ini ke dalam parameter pertama fungsi validate(). baru kemudian kita buat pengaturan validasi nya masing-masing.</p>

	<p>Pada contoh form validasi laravel sederhana ini kita membuat pengaturan ‘required’ pada form nama, nim, email dan jurusan. yang mana artinya form ini wajib untuk di isi (tidak boleh kosong). Jika validasi berhasil, atau jika yang di input sesuai dengan pengaturan yang sudah kita tetapkan dalam fungsi validate() tadi, maka kita passing data yang di input oleh user ke view <code>simpan.blade.php</code></p>																
	<p>Berikut beberapa validasi yang sering dibutuhkan.</p> <table> <tr> <td>required</td><td>membuat form wajib isi</td></tr> <tr> <td>min:10</td><td>membuat minimal karakter yang diinput</td></tr> <tr> <td>max:10</td><td>membuat maksimal karakter yang boleh diinput</td></tr> <tr> <td>alpha</td><td>hanya boleh menginput huruf</td></tr> <tr> <td>numeric</td><td>hanya boleh menginput angka</td></tr> <tr> <td>alpha_num</td><td>hanya boleh menginput angka dan huruf (!@#\$%^&* dan lain-lain tidak diperbolehkan)</td></tr> <tr> <td>bail</td><td>berhenti menjalankan validasi selanjutnya jika validasi sebelumnya gagal (error)</td></tr> <tr> <td>email</td><td>hanya boleh menginput email</td></tr> </table>	required	membuat form wajib isi	min:10	membuat minimal karakter yang diinput	max:10	membuat maksimal karakter yang boleh diinput	alpha	hanya boleh menginput huruf	numeric	hanya boleh menginput angka	alpha_num	hanya boleh menginput angka dan huruf (!@#\$%^&* dan lain-lain tidak diperbolehkan)	bail	berhenti menjalankan validasi selanjutnya jika validasi sebelumnya gagal (error)	email	hanya boleh menginput email
required	membuat form wajib isi																
min:10	membuat minimal karakter yang diinput																
max:10	membuat maksimal karakter yang boleh diinput																
alpha	hanya boleh menginput huruf																
numeric	hanya boleh menginput angka																
alpha_num	hanya boleh menginput angka dan huruf (!@#\$%^&* dan lain-lain tidak diperbolehkan)																
bail	berhenti menjalankan validasi selanjutnya jika validasi sebelumnya gagal (error)																
email	hanya boleh menginput email																
4	Buat view tambah di <code>laravel-crud/resources/views/tambah.blade.php</code>																

```


resources > views > tambah.blade.php > form > div.form-group > label
1  @extends('master')
2
3  <!--isi file-->
4  @section('title','Tambah Data')
5
6  <!--isi bagian judul halaman-->
7  @section('judul_halaman','Tambah Data Mahasiswa')
8
9  <!--isi bagian konten-->
10 @section('konten')
11 <a href="/mahasiswa" class="btn btn-danger">Kembali</a>
12 <br />
13 <br />
14 <!--Menampilkan error valid-->
15 @if (count($errors)>0)
16 <div class="alert alert-danger">
17     <ul>
18         @foreach ($errors->all() as $error)
19             <li>{{ $error }}</li>
20         @endforeach
21     </ul>
22 </div>
23 @endif
24 <!--form validasi-->
25 <form action="/mahasiswa/simpan" method="post">
26     {{csrf_field()}}
27     <div class="form-group">
28         <label for="nama">Nama</label>
29         <input class="form-control" type="text" name="nama" value="{{old('nama')}}"><br />
30     </div>
31     <div class="form-group">
32         <label for="nim">NIM</label>
33         <input class="form-control" type="text" name="nim" value="{{old('nim')}}"><br />
34     </div>
35     <div class="form-group">
36         <label for="email">Email</label>
37         <input class="form-control" type="text" name="email" value="{{old('email')}}"><br />
38     </div>
39     <div class="form-group">
40         <label for="jurusan">Jurusan</label>
41         <input class="form-control" type="text" name="jurusan" value="{{old('jurusan')}}"><br />
42     </div>
43     <div class="form-group">
44         <button type="submit" name="tambah" class="btn btn-primary float-right">Tambah Data</button>
45     </div>
46 </form>
47 @endsection

```

pada form ini kita mengarahkan ke route '/mahasiswa/simpan' dengan method post.

perhatikan pada value nya masing-masing. ada fitur keren dari laravel. yaitu fungsi old(). fungsi ini berguna untuk menampilkan data yang sebelumnya di input. Jadi jika ada validasi error, maka data yang di isi pada form tidak hilang, tapi tetap masih ditampilkan.

	<p>karena tentu akan sangat tidak efektif jika user harus menginput data berulang-ulang jika ada validasi yang salah pada penginputan.</p> <p>Di sana kita juga sudah membuat pengecekan jika ada error dari validasi form nya, maka kita tampilkan errornya secara keseluruhan dan berurut. jika jumlah error lebih besar dari 0 (ada), maka tampilkan semua error dengan menggunakan fungsi foreach(). Lihat baris 12-22</p>
5	<p>Selanjutnya buat view simpan di laravel-crud/resources/views/simpan.blade.php</p> <pre> resources > views >  simpan.blade.php > ... 1 @extends ('master') 2 3 <!--isi file--> 4 @section ('title', 'Validasi Data') 5 6 <!--isi bagian judul halaman--> 7 @section ('judul_halaman', 'Validasi Data Mahasiswa') 8 9 <!-- isi bagian konten--> 10 @section ('konten') 11 <h3 class="my-4">Data Yang Di Input </h3> 12 13 <table class="table table-bordered table-striped"> 14 <tr> 15 <td style="width:150px">Nama</td> 16 <td>{{ \$data->nama }}</td> 17 </tr> 18 <tr> 19 <td>NIM</td> 20 <td>{{ \$data->nim }}</td> 21 </tr> 22 <tr> 23 <td>Email</td> 24 <td>{{ \$data->email }}</td> 25 </tr> 26 <tr> 27 <td>Jurusan</td> 28 <td>{{ \$data->jurusan }}</td> 29 </tr> 30 </table> 31 32 Kembali 33 @endsection </pre> <p>Pada view simpan.blade.php ini kita tampilkan data-data yang dikirim dari method input() tadi.</p>
6	<p>Jalankan Project Laravel Kita</p>

	<pre>C:\xampp\htdocs\laravel-crud>php artisan serve Laravel development server started: http://127.0.0.1:8000</pre>
7	<p>Akses http://127.0.0.1:8000/</p> 
8	<p>Login sesuai dengan user yang telah dibuat pada pertemuan sebelumnya.</p>  <p>Laravel</p> <ul style="list-style-type: none"> • Login • Register <p>Login</p> <p>E-Mail Address</p> <p><input type="text" value="silviindah23@gmail.com"/></p> <p>Password</p> <p><input type="password" value="....."/></p> <p><input type="checkbox"/> Remember Me</p> <p><input type="button" value="Login"/> Forgot Your Password?</p>
9	<p>Selanjutnya buka http://127.0.0.1:8000/mahasiswa dan Klik tombol Tambah Data Mahasiswa, lalu masukkan data mahasiswa yang tidak lengkap .</p>

Data Mahasiswa

[Tambah Data Mahasiswa](#)

Nama	NIM	
nanda eka	1001001001	Detail Edit Hapus
wahyu	1001001002	Detail Edit Hapus

Tambah data mahasiswa tidak lengkap

Tambah Data Mahasiswa

[Kembali](#)

Nama
NIM
Email
Jurusan

10 Setelah Klik tombol tambah dan hasilnya akan seperti ini

Tambah Data Mahasiswa

[Kembali](#)

- The email field is required.

Nama
NIM
Email
Jurusan

11 Masukkan data mahasiswa yang benar, dan hasilnya akan seperti ini.

	<div data-bbox="321 216 852 264" data-label="Section-Header"> <h2>Tambah Data Mahasiswa</h2> </div> <div data-bbox="319 304 444 344" data-label="Text"> <p>Kembali</p> </div> <div data-bbox="319 380 776 588" data-label="Form"> <p>Nama <input type="text" value="Risky Adi Purnama"/></p> <p>NIM <input type="text" value="1841720456"/></p> <p>Email <input type="text" value="riskyadi@gmail.com"/></p> <p>Jurusan <input type="text" value="Teknik Informatika"/></p> <p><input type="button" value="Tambah Data"/></p> </div>
12	<p>Data mahasiswa berhasil ditambahkan, kita dapat mengecek data tersebut pada halaman http://127.0.0.1:8000/mahasiswa</p>