

LAPORAN PEMROGRAMAN WEB LANJUT

JOBSHEET 13

MEMBUAT FORM VALIDASI MENGGUNAKAN LARAVEL

Oleh:

KINANTI PERMATA PUTRI

NIM. 1841720022



PROGRAM STUDI TEKNIK INFORMATIKA

JURUSAN TEKNOLOGI INFORMASI

POLITEKNIK NEGERI MALANG

APRIL 2020

A. Praktikum: Membuat Form Validasi di Laravel

- 1) Buka project Laravel “laravel-crud”
- 2) 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.

```
27
28 Route::get('/mahasiswa/tambah', 'MahasiswaController@tambah');
29
30 Route::post('/mahasiswa/simpan', 'MahasiswaController@simpan');
```

- 3) Buat 2 buah method pada laravel-crud/app/Http/controller/MahasiswaController.php.

```
24 public function tambah()
25 {
26     // Memanggil view tambah
27     return view('tambah');
28 }
29
30 public function simpan(Request $request)
31 {
32     $this->validate($request, [
33         'nama' => 'required',
34         'nim' => 'required|numeric',
35         'email' => 'required|email',
36         'jurusan' => 'required'
37     ]);
38
39     // Insert data ke tabel mahasiswa
40     DB::table('mahasiswa')->insert([
41         'nama' => $request->nama,
42         'nim' => $request->nim,
43         'email' => $request->email,
44         'jurusan' => $request->jurusan,
45     ]);
46     return view('simpan', ['data' => $request]);
47 }
```

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.

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.

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 simpan.blade.php

Berikut beberapa validasi yang sering dibutuhkan.

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 laravel-crud/resources/views/tambah.blade.php

```
1  @extends('master')
2  <!-- Isi title -->
3  @section('title', 'Tambah Data')
4
5  <!-- Isi bagian judul halaman -->
6  @section('judul_halaman', 'Tambah Data Mahasiswa')
7
8  <!-- Isi bagian konten -->
9  @section('konten')
10     <a href="/mahasiswa" class="btn btn-danger">Kembali</a>
11     <br/>
12     <br/>
13     <!-- Menampilkan error validasi -->
14     @if (count($error) > 0)
15         <div class="alert alert-danger">
16             <ul>
17                 @foreach ($error->all() as $error)
18                     <li>{{ $error }}</li>
19                 @endforeach
20             </ul>
21         </div>
22     @endif
```

```

22 @endif
23 <!-- Form validasi -->
24 <form action="/mahasiswa/simpan" method="post">
25     {{ csrf_field() }}
26
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">E-mail</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         <input class="btn btn-primary" type="submit" value="Tambah">
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. karena tentu akan sangat tidak efektif jika user harus menginput data berulang-ulang jika ada validasi yang salah pada penginputan.

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

5) Selanjutnya buat view simpan di laravel-crud/resources/views/simpan.blade.php

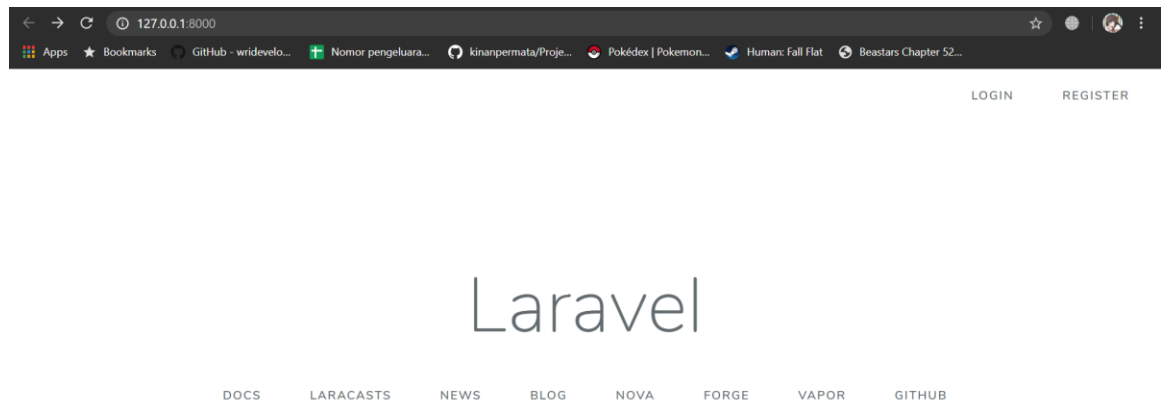
```
1  @extends('master')
2
3  <!-- Isi title -->
4  @section('title', 'Validasi Data')
5
6  <!-- Isi bagian judul halaman -->
7  @section('judul_halaman', 'Validasi Data Mahasiswa')
8
9  <!-- Isi bagian konten-->
10 @section('konten')
11 <h3 class="my-4">Data yang diinput : </h3>
12
13     <table class="table 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     <a href="/mahasiswa" class="btn btn-primary">Kembali</a>
33 @endsection
```

Pada view simpan.blade.php ini kita tampilkan data-data yang dikirim dari method input() tadi.

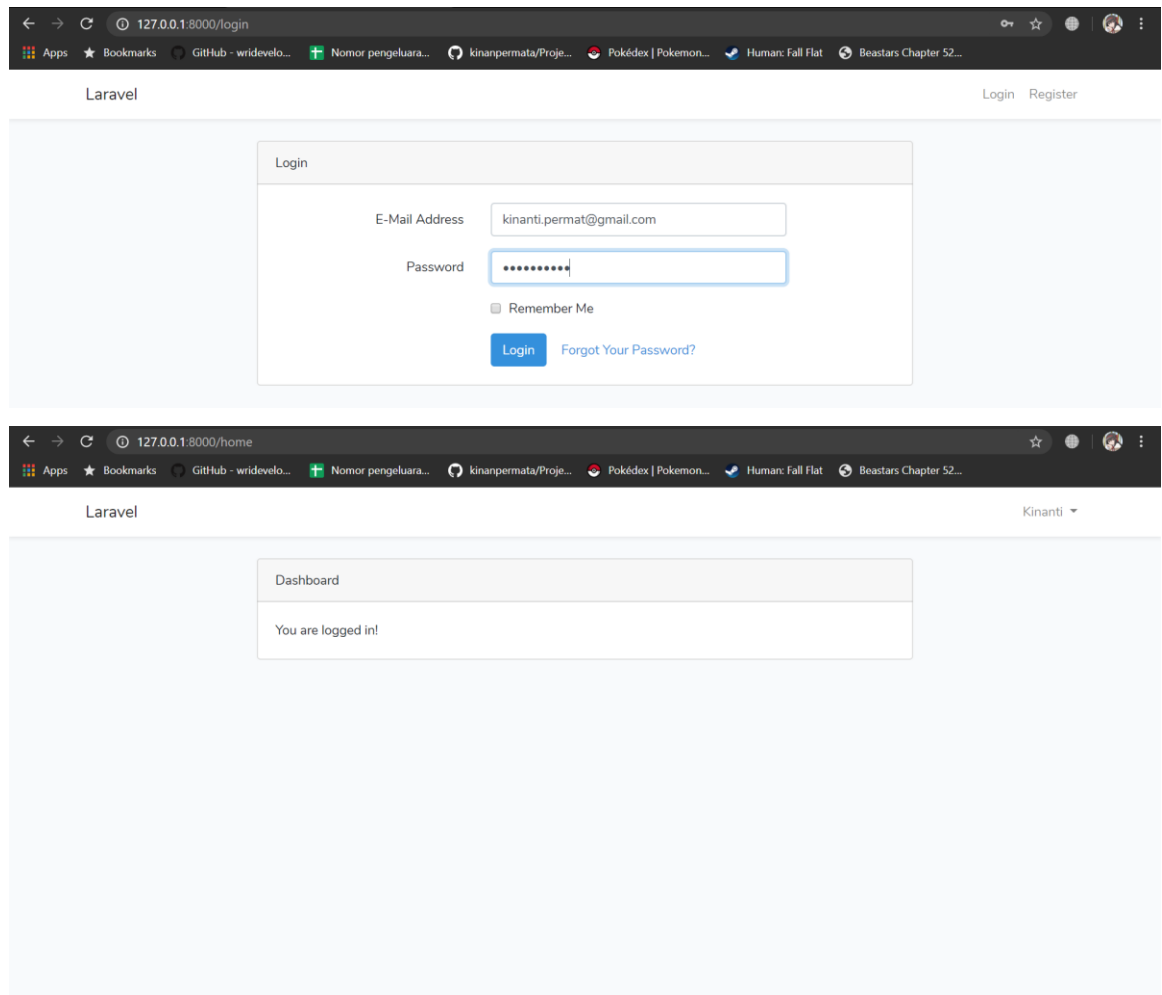
6) Jalankan Project Laravel Kita

```
C:\xampp\htdocs\laravel-crud>php artisan serve
Laravel development server started: http://127.0.0.1:8000
```

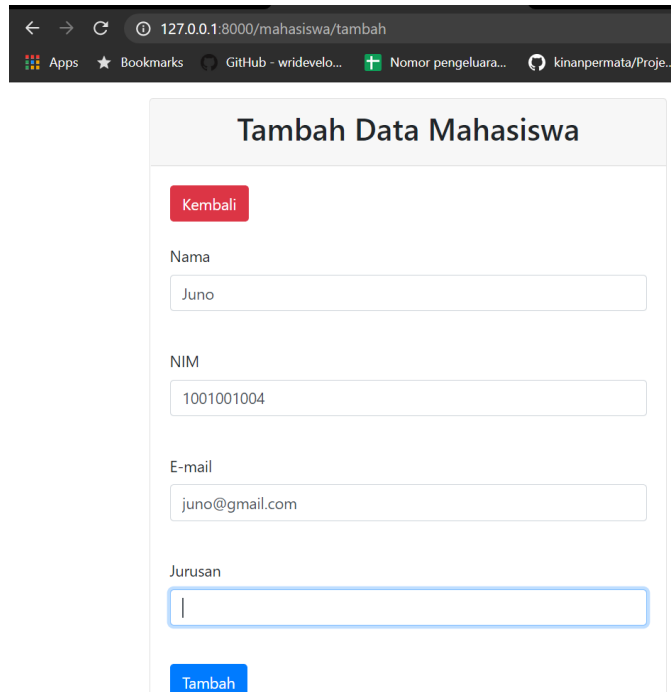
7) Akses <http://127.0.0.1:8000/>



8) Login sesuai dengan user yang telah dibuat pada pertemuan sebelumnya.



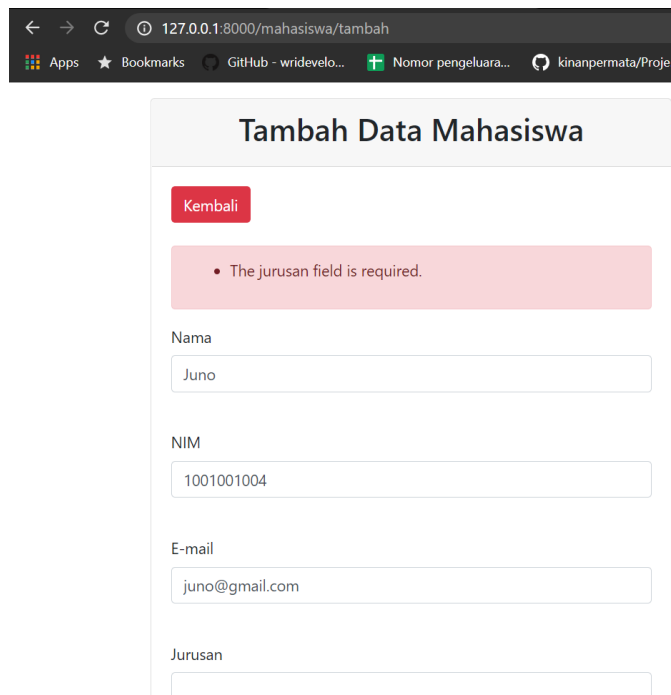
- 9) Selanjutnya buka [http://127.0.0.1:8000/mahasiswa](http://127.0.0.1:8000/mahasiswa/tambah) dan Klik tombol Tambah Data Mahasiswa, lalu masukkan data mahasiswa yang tidak lengkap.



The screenshot shows a web browser window with the address bar displaying `127.0.0.1:8000/mahasiswa/tambah`. The page title is "Tambah Data Mahasiswa". The form contains the following fields and values:

- Kembali** (button)
- Nama**: Juno
- NIM**: 1001001004
- E-mail**: juno@gmail.com
- Jurusan**: (empty field)
- Tambah** (button)

- 10) Setelah itu Klik tombol tambah dan hasilnya akan seperti ini



The screenshot shows the same web browser window, but now a red error message is displayed above the "Jurusan" field:

- The jurusan field is required.**

The form fields and buttons remain the same as in the previous screenshot.

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

← → ↻ 127.0.0.1:8000/mahasiswa/tambah

Apps ★ Bookmarks GitHub - widevelo... Nomor pengeluaran... kinanpermata/Proje...

Kembali

• The jurusan field is required.

Nama

Juno

NIM

1001001004

E-mail

juno@gmail.com

Jurusan

Teknik Mesin

Tambah

← → ↻ 127.0.0.1:8000/mahasiswa/simpan

Apps ★ Bookmarks GitHub - widevelo... Nomor pengeluaran... kinanpermata/Proje...

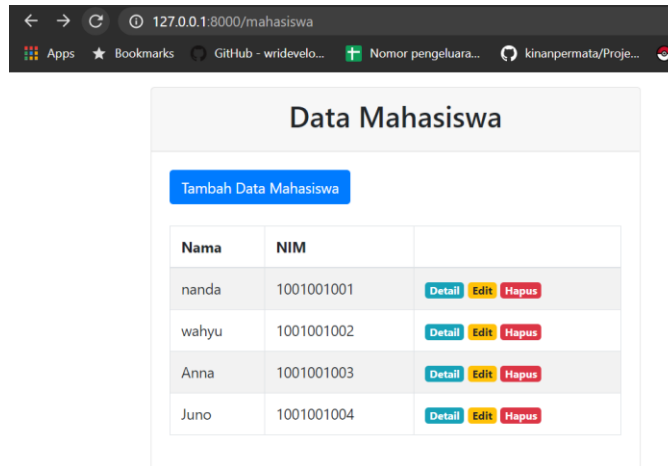
Validasi Data Mahasiswa

Data yang diinput :

Nama	Juno
NIM	1001001004
Email	juno@gmail.com
Jurusan	Teknik Mesin

Kembali

12) Data mahasiswa berhasil ditambahkan, kita dapat mengecek data tersebut pada halaman <http://127.0.0.1:8000/mahasiswa>



The screenshot shows a web browser window with the address bar displaying `127.0.0.1:8000/mahasiswa`. The browser's bookmark bar includes links to 'Apps', 'Bookmarks', 'GitHub - widevelo...', 'Nomor pengeluaran...', and 'kinanpermata/Proje...'. The main content area of the browser shows a web application titled 'Data Mahasiswa'. At the top of the application is a blue button labeled 'Tambah Data Mahasiswa'. Below this button is a table with the following data:

Nama	NIM	
nanda	1001001001	Detail Edit Hapus
wahyu	1001001002	Detail Edit Hapus
Anna	1001001003	Detail Edit Hapus
Juno	1001001004	Detail Edit Hapus