



Jurusan Teknologi Informasi Politeknik Negeri Malang
Jobsheet-13: Membuat Form Validasi Menggunakan Laravel

Mata Kuliah Pemrograman Web Lanjut

Pengampu: Tim Ajar Pemrograman Web Lanjut

April 2020

NAMA : HUSNUL HOTIMAH

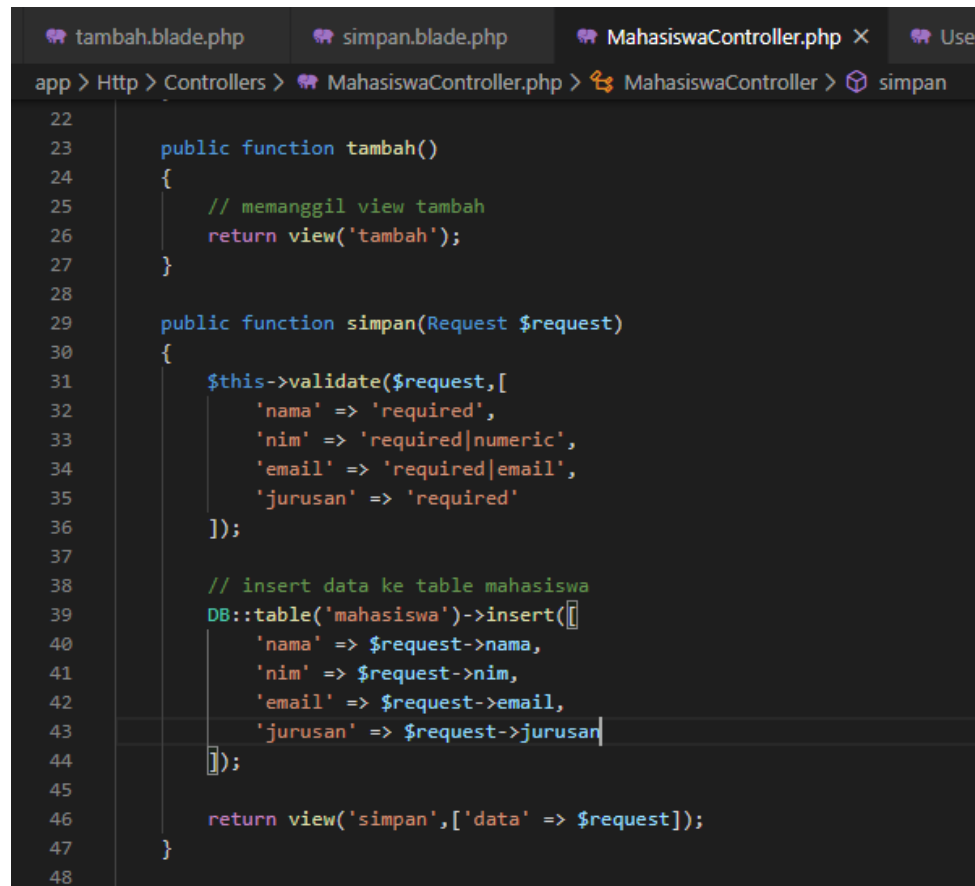
NIM : 1841720014

KELAS : TI-2A

Praktikum: Membuat Form Validasi di Laravel

Langkah	Keterangan
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>web.php x app.blade.php index.blade.php ...users index.blade.php routes > web.php > ... 22 23 Route::get('/mahasiswa/tambah', 'MahasiswaController@tambah'); 24 Route::post('/mahasiswa/simpan', 'MahasiswaController@simpan'); 25 26</pre>

3

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


```

22
23     public function tambah()
24     {
25         // memanggil view tambah
26         return view('tambah');
27     }
28
29     public function simpan(Request $request)
30     {
31         $this->validate($request,[
32             'nama' => 'required',
33             'nim' => 'required|numeric',
34             'email' => 'required|email',
35             'jurusan' => 'required'
36         ]);
37
38         // insert data ke table mahasiswa
39         DB::table('mahasiswa')->insert([
40             'nama' => $request->nama,
41             'nim' => $request->nim,
42             'email' => $request->email,
43             'jurusan' => $request->jurusan
44         ]);
45
46         return view('simpan',['data' => $request]);
47     }
48

```

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.

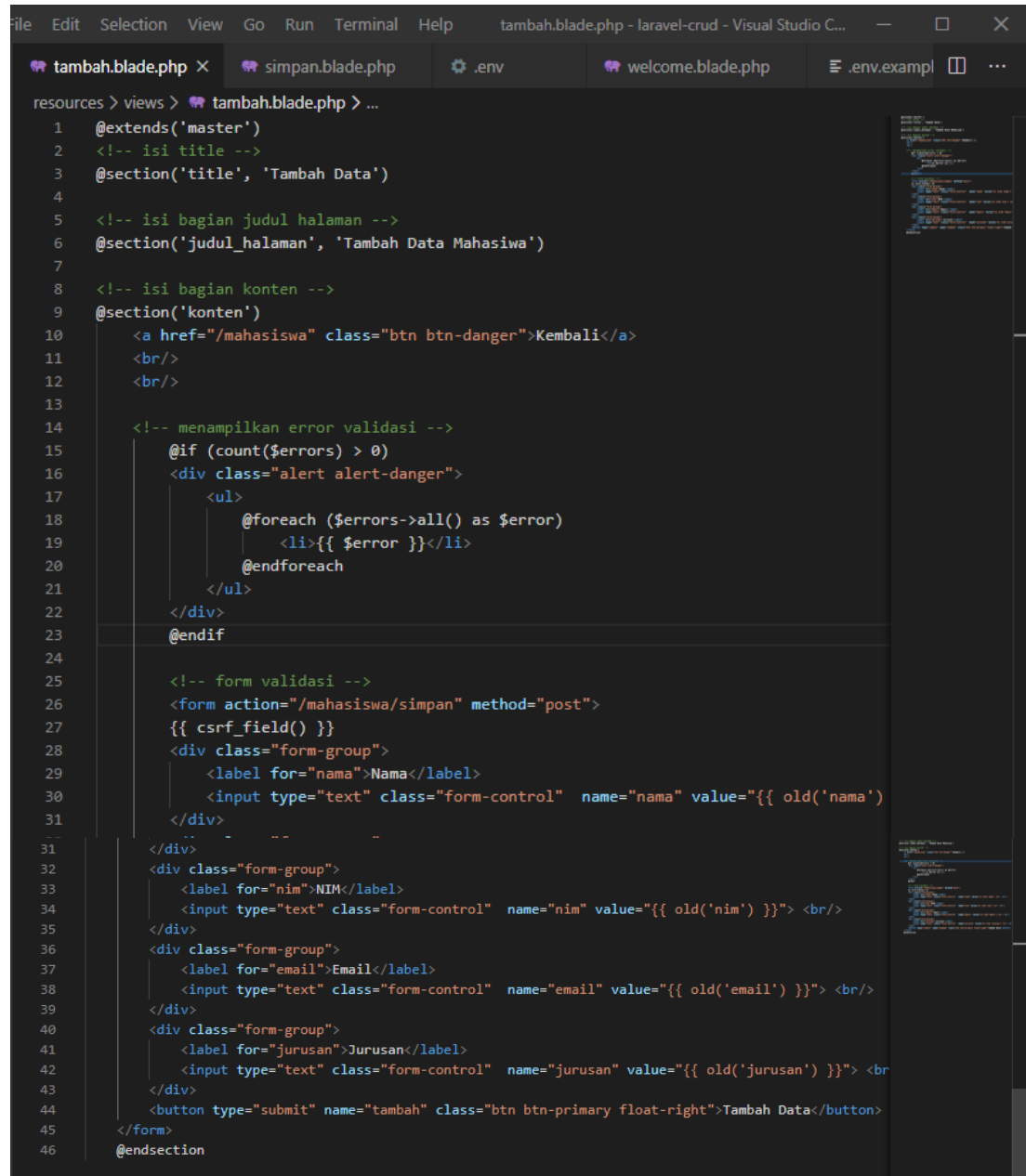
Pada contoh form validasi laravel sederhana ini kita membuat pengaturan **'required'** pada form nama, nim, email dan jurusan. yang mana artinya form ini wajib untuk di isi (tidak boleh kosong). Jika validasi berhasil, atau jika yang di input sesuai dengan pengaturan yang sudah kita tetapkan dalam fungsi `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



```

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
11     <a href="/mahasiswa" class="btn btn-danger">Kembali</a>
12     <br/>
13     <br/>
14
15     <!-- menampilkan error validasi -->
16     @if (count($errors) > 0)
17         <div class="alert alert-danger">
18             <ul>
19                 @foreach ($errors->all() as $error)
20                     <li>{{ $error }}</li>
21                 @endforeach
22             </ul>
23         </div>
24     @endif
25
26     <!-- form validasi -->
27     <form action="/mahasiswa/simpan" method="post">
28         {{ csrf_field() }}
29         <div class="form-group">
30             <label for="nama">Nama</label>
31             <input type="text" class="form-control" name="nama" value="{{ old('nama') }}">
32         </div>
33         <div class="form-group">
34             <label for="nim">NIM</label>
35             <input type="text" class="form-control" name="nim" value="{{ old('nim') }}"> <br/>
36         </div>
37         <div class="form-group">
38             <label for="email">Email</label>
39             <input type="text" class="form-control" name="email" value="{{ old('email') }}"> <br/>
40         </div>
41         <div class="form-group">
42             <label for="jurusan">Jurusan</label>
43             <input type="text" class="form-control" name="jurusan" value="{{ old('jurusan') }}"> <br/>
44         </div>
45         <button type="submit" name="tambah" class="btn btn-primary float-right">Tambah Data</button>
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

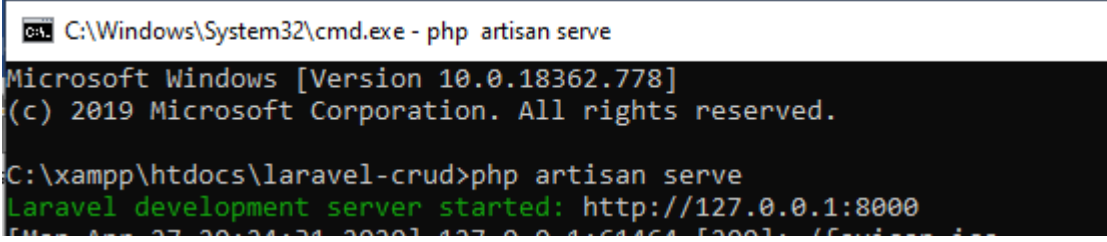
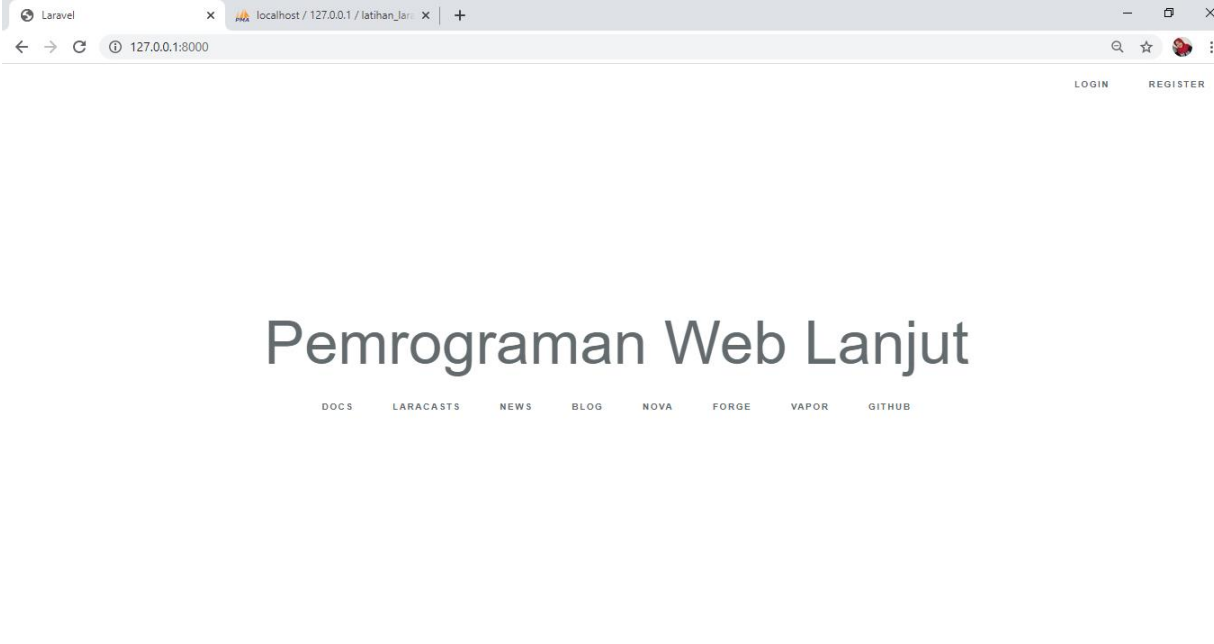
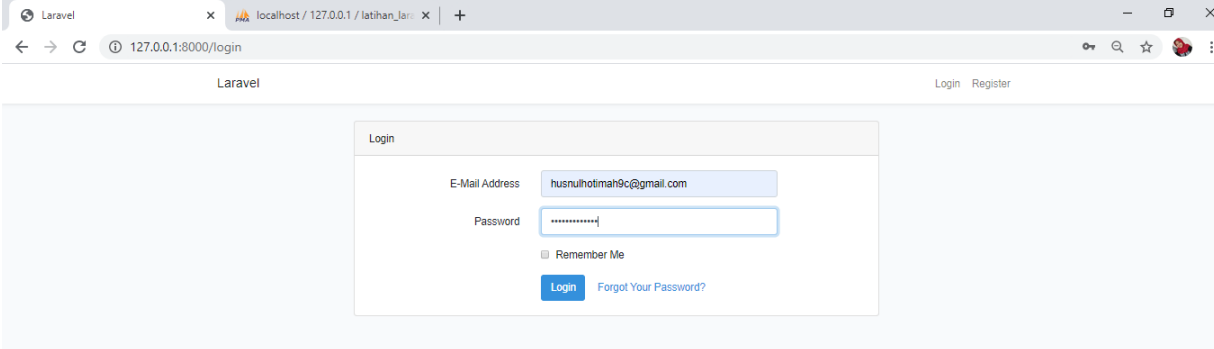
Selanjutnya buat view simpan di `laravel-crud/resources/views/simpan.blade.php`

```
tambah.blade.php  simpan.blade.php X  index.blade.php  MahasiswaController  ...

resources > views > simpan.blade.php > ...

1  @extends('master')
2  <!-- isi title -->
3  @section('title', 'Validasi Data')
4
5  <!-- isi bagian judul halaman -->
6  @section('judul_halaman', 'Validasi Data Mahasiwa')
7
8  <!-- isi bagian konten -->
9  @section('konten')
10 <h3 class="my-4">Data Yang di Input : </h3>
11
12     <table class="table table-bordered table-striped">
13         <tr>
14             <td style="width:150px">Nama</td>
15             <td>{{ $data->nama }}</td>
16         </tr>
17         <tr>
18             <td>NIM</td>
19             <td>{{ $data->nim }}</td>
20         </tr>
21         <tr>
22             <td>Email</td>
23             <td>{{ $data->email }}</td>
24         </tr>
25         <tr>
26             <td>Jurusan</td>
27             <td>{{ $data->jurusan }}</td>
28         </tr>
29     </table>
30
31     <a href="/mahasiswa" class="btn btn-primary">Kembali</a>
32
33
34 @endsection
```

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

6	<p>Jalankan Project Laravel Kita</p>  <pre> C:\Windows\System32\cmd.exe - php artisan serve Microsoft Windows [Version 10.0.18362.778] (c) 2019 Microsoft Corporation. All rights reserved. C:\xampp\htdocs\laravel-crud>php artisan serve Laravel development server started: http://127.0.0.1:8000 [Mon Apr 27 20:34:34 2020] 127.0.0.1:61451 [500] - /index.php </pre>
7	<p>Akses http://127.0.0.1:8000/</p> 
8	<p>Login sesuai dengan user yang telah dibuat pada pertemuan sebelumnya.</p> 

9

Selanjutnya buka <http://127.0.0.1:8000/mahasiswa> dan Klik tombol Tambah Data Mahasiswa, lalu masukkan data mahasiswa yang tidak lengkap .

The image shows two screenshots of a web application running on a local host.

The top screenshot displays the "Data Mahasiswa" page. It features a table with student data and a "Tambah Data Mahasiswa" button.

Nama	NIM	
Husnul Hotimah	1841720014	Detail Edit Hapus
Zaskia Sungkar	1841720012	Detail Edit Hapus
Muhammad Amar	1841720013	Detail Edit Hapus
Ahmad Zidan	1841720021	Detail Edit Hapus

The bottom screenshot displays the "Tambah Data Mahasiswa" form. It includes a "Kembali" button and input fields for Nama, NIM, Email, and Jurusan.

Kembali

Nama
Farida

NIM
1841720123

Email

Jurusan
Akuntansi

Tambah Data

10

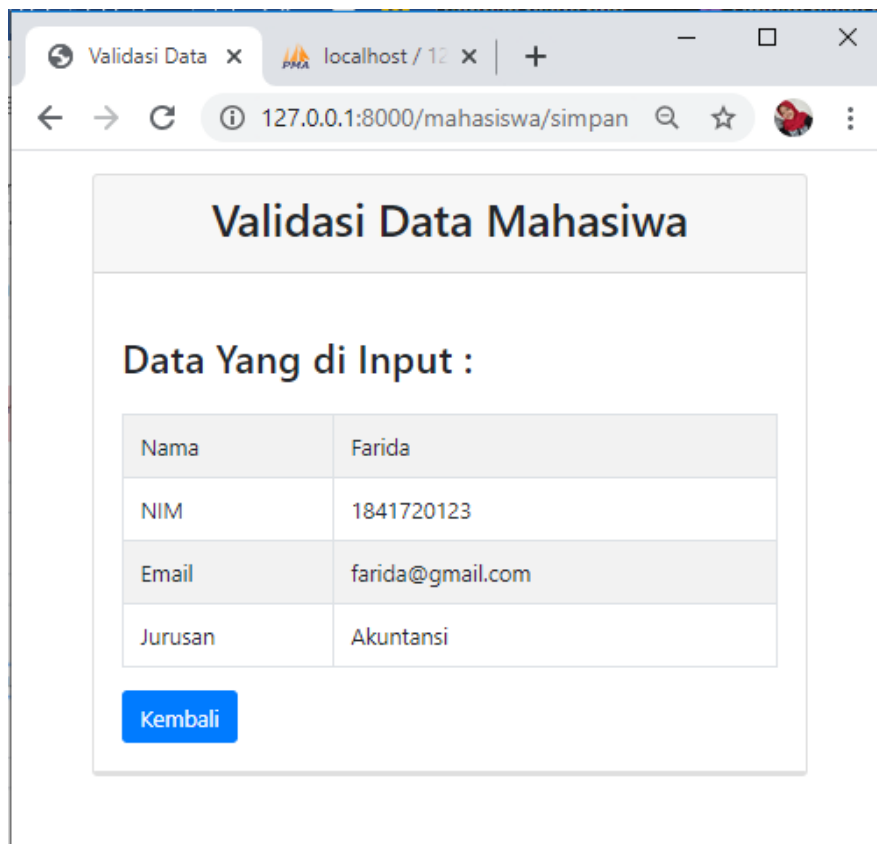
Setelah itu Klik tombol tambah dan hasilnya akan seperti ini

The screenshot shows a web browser with two tabs: 'Tambah Data' and 'localhost / 127.0.0.1 / latihan_jar...'. The address bar shows '127.0.0.1:8000/mahasiswa/tambah'. The form titled 'Tambah Data Mahasiwa' contains a red 'Kembali' button at the top. Below it is a red error message box stating 'The email field is required.' The form fields are: 'Nama' (Farida), 'NIM' (1841720123), 'Email' (empty), and 'Jurusan' (Akuntansi). A blue 'Tambah Data' button is at the bottom right.

11

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

The screenshot shows the same web browser and form as in step 10. The 'Email' field now contains 'farida@gmail.com' and is highlighted with a blue border. The 'Tambah Data' button remains at the bottom right.



12

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

