




LAPORAN PRAKTIKUM PEMROGRAMAN DASAR WEB

NAMA : NUR HANIFAH
NIM : 1841720062
Kelas : TI 2B
Isi Laporan : Jobsheet9 - Pengantar dan Instalasi Laravel Mata Kuliah Pemrograman

Praktikum – Bagian 1: Instalasi Laravel di Windows

Langkah	Keterangan
1	<p>Laravel menggunakan Composer untuk mengelola dependensinya. Jadi, sebelum menggunakan Laravel, pastikan Composer telah terinstall terlebih dahulu. Composer adalah dependensi manajer untuk library PHP. Composer digunakan untuk menginstal, menghapus, dan memperbarui paket PHP. Unduh composer melalui https://getcomposer.org/, pilih Download untuk memulai proses unduh.</p> 
2	<p>Terdapat 2 cara dalam menginstall Laravel,</p> <p>1. Install Via Laravel Installer</p> <p>Setelah proses instalasi Composer selesai, buka Command Prompt lalu tujulah lokasi folder yang akan menampung folder kerja kita. Karena pada praktikum sebelumnya telah</p>

menggunakan XAMPP, maka dapat menggunakan folder htdocs. Pindah ke direktori htdocs dengan ketik: **cd C:\xampp\htdocs**. Setelah berada di dalam folder, ketikkan perintah sebagai berikut:

composer global require "laravel/installer"

Tunggulah hingga proses penginstalan selesai. Setelah itu, Kita bisa mulai membuat project laravel dengan mengetikkan perintah:

laravel new laravelapp

Keterangan:

laravelapp: merupakan folder kerja laravel, Kita dapat menggantinya sesuai dengan keinginan

2. Install Via Composer Create-Project

Selain menggunakan cara pertama, Kita dapat melakukan penginstalan laravel dengan mengetikkan perintah berikut pada command prompt:

composer create-project --prefer-dist laravel/laravel laravelapp

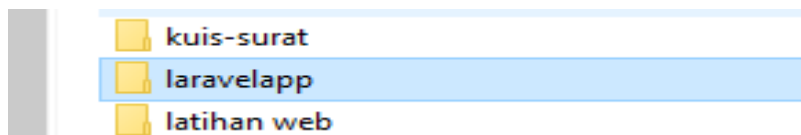
Contoh Pengerjaan:

```
Microsoft Windows [Version 10.0.18363.720]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\ASUS A405U>cd C:\xampp\htdocs

C:\xampp\htdocs>composer create-project --prefer-dist laravel/laravel laravelapp
Installing laravel/laravel (v7.3.0)
- Installing laravel/laravel (v7.3.0): Downloading (100%)
Created project in laravelapp
> @php -r "file_exists('.env') || copy('.env.example', '.env');"
Loading composer repositories with package information
Updating dependencies (including require-dev)
Package operations: 94 installs, 0 updates, 0 removals
- Installing voku/portable-ascii (1.4.10): Downloading (100%)
- Installing symfony/polyfill-ctype (v1.15.0): Downloading (100%)
- Installing phoption/phoption (1.7.3): Downloading (100%)
- Installing vlucas/phpdotenv (v4.1.3): Downloading (100%)
- Installing symfony/css-selector (v5.0.7): Downloading (connecting...)
```

Setelah terinstall semua pastikan laravelapp sudah ada di dalam htdocs



3

Setelah proses instalasi Laravel selesai, Kita perlu menguji apakah hasil instalasi tersebut berjalan dengan baik atau tidak. Aktifkan Apache server lewat XAMPP Control Panel. Pada browser, ketik alamat <http://localhost/laravelapp/public/> Atau Kita dapat menggunakan perintah berikut di Command Prompt:

php artisan serve

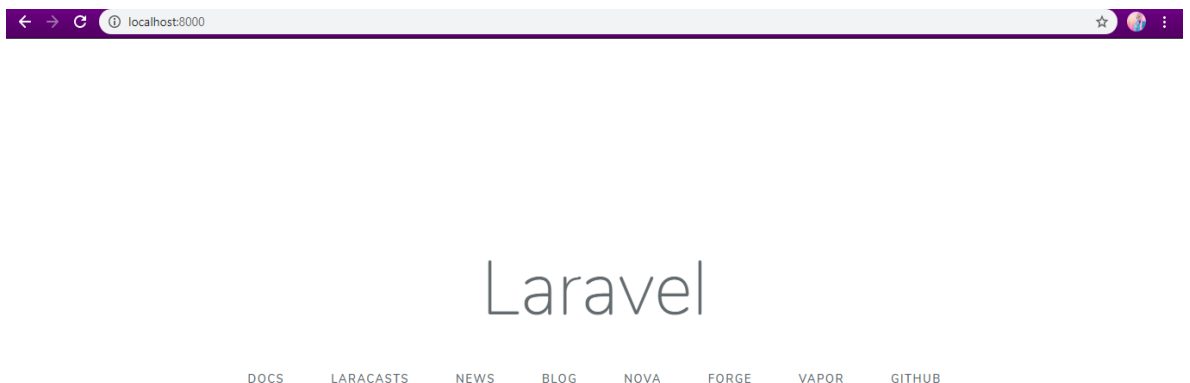
Contoh:

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

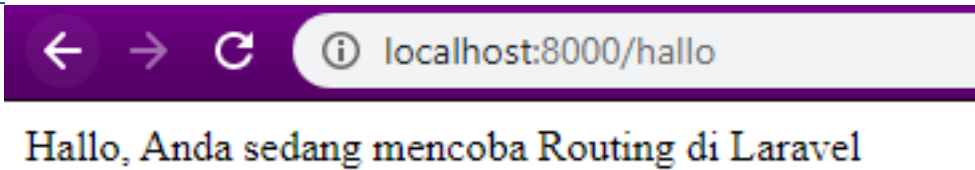
Secara default, Anda akan diarahkan menuju alamat server, yaitu 127.0.0.1:8000.

Kita dapat menjalankan aplikasi LARAVEL di folder manapun dengan menggunakan perintah di atas sehingga tidak harus selalu berada di dalam folder htdocs (XAMPP).

Jalankan dengan localhost:8000, dan pastikan tampilan di browser akan seperti berikut ini :



Praktikum – Bagian 2: Mengenal dan membuat route

Langkah	Keterangan
1	<p>Setiap request yang datang pada laravel akan diarahkan melalui sebuah route. Route ini yang akan menentukan respon apa yang akan dikerjakan untuk membalas request tersebut. Halaman homepage default Laravel seperti yang ditampilkan pada gambar di Bagian 1 adalah hasil dari script yang terletak pada file laravelapp\routes\web.php Berikut ini adalah perintah route untuk menampilkan halaman default Laravel tersebut</p> <pre>web.php x routes > web.php > ... 1 <?php 2 3 use Illuminate\Support\Facades\Route; 4 5 /* 6 ----- 7 Web Routes 8 ----- 9 10 Here is where you can register web routes for your application. These 11 routes are loaded by the RouteServiceProvider within a group which 12 contains the "web" middleware group. Now create something great! 13 14 */ 15 16 Route::get('/', function () { 17 return view('welcome'); 18 }); 19</pre>
2	<p>Kita dapat menuliskan perintah baru untuk membuat route, selain mengembalikan file view sebagai response, Kita dapat mengembalikan response berupa string atau teks biasa. Buka file laravelapp\routes\web.php, tambahkan isinya dengan script di bawah ini :</p> <p>Route::get('hallo', function () { return "Hallo, Anda sedang mencoba Routing di Laravel"; });</p> <p>Untuk melihat hasilnya, pada browser, ketikkan alamat http://localhost:8000/hallo</p> <p>Hasilnya:</p> 

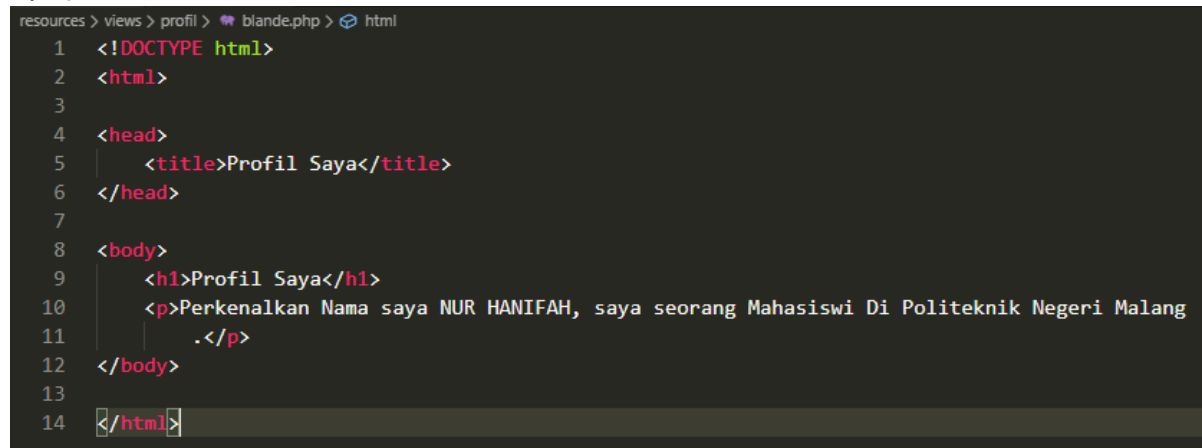
3

Selanjutnya, kita akan mencoba membuat route baru dengan menampilkan halaman view. Tambahkan route baru untuk halaman “profil” dengan menambahkan script berikut:

Route::get('profil', function () { return view('profil'); });

Buat file baru bernama profil.blade.php di dalam folder laravelapp\resources\views\, ketikkan script berikut ini:

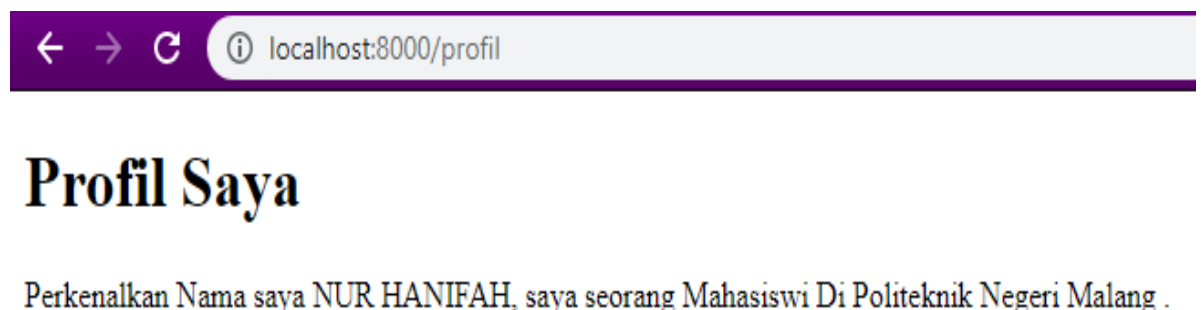
```
<!DOCTYPE html>
<html>
<head>
    <title>Profil Saya</title>
</head>
<body>
    <h1>Profil Saya</h1>
    <p>Perkenalkan Nama saya nama_saya, saya seorang .....
.</p>
</body>
</html>
```



```
resources > views > profil > blade.php > html
1 <!DOCTYPE html>
2 <html>
3
4 <head>
5     <title>Profil Saya</title>
6 </head>
7
8 <body>
9     <h1>Profil Saya</h1>
10    <p>Perkenalkan Nama saya NUR HANIFAH, saya seorang Mahasiswi Di Politeknik Negeri Malang
11    .</p>
12 </body>
13
14 </html>
```

Tuliskan profil kalian masing-masing. Untuk menampilkan hasilnya, ketikkan alamat <http://localhost:8000/profil>

HASIL:



← → ↻ ⓘ localhost:8000/profil

Profil Saya

Perkenalkan Nama saya NUR HANIFAH, saya seorang Mahasiswi Di Politeknik Negeri Malang .

Praktikum – Bagian 3: Cara Menggunakan Controller pada Framework Laravel

Langkah	Keterangan
1	<p>Pada pembahasan sebelumnya, Kita sudah bisa menampilkan file view langsung dari Routenya. Tetapi sebuah aplikasi tidaklah sesimpel itu, akan ada logika dan data-data yang harus diolah terlebih dahulu. Caranya adalah dengan menggunakan Controller, seperti yang telah dipelajari ketika menggunakan CodeIgniter.</p> <p>Ada dua cara yang dapat digunakan untuk membuat Controller. Cara pertama, dengan membuat file controller secara manual dan tuliskan code untuk extends controller secara manual. Cara kedua, adalah dengan membuat file Controller lewat Artisan di Laravel.</p> <p>Kita gunakan cara yang kedua, pada latihan kali ini kita akan membuat Controller dengan nama CobaController dengan menuliskan di command prompt / terminal :</p> <p style="text-align: center;">php artisan make:controller CobaController</p> <p>Contoh:</p> <pre>Microsoft Windows [Version 10.0.18363.720] (c) 2019 Microsoft Corporation. All rights reserved. C:\Users\ASUS A405U>cd C:\xampp\htdocs C:\xampp\htdocs>cd laravelapp C:\xampp\htdocs\laravelapp>php artisan make:controller CobaController Controller created successfully. C:\xampp\htdocs\laravelapp></pre> <p>Maka akan terbentuk sebuah file dengan nama CobaController.php pada folder laravelapp\app\Http\Controllers\</p> <pre>app > Http > Controllers > CobaController.php > ... 1 <?php 2 3 namespace App\Http\Controllers; 4 5 use Illuminate\Http\Request; 6 7 class CobaController extends Controller 8 { 9 // 10 } 11</pre>
2	<p>Ubah route halaman 'profil' pada praktikum sebelumnya menjadi seperti berikut</p> <p style="text-align: center;">Route::get('profil', 'CobaController@profil');</p>

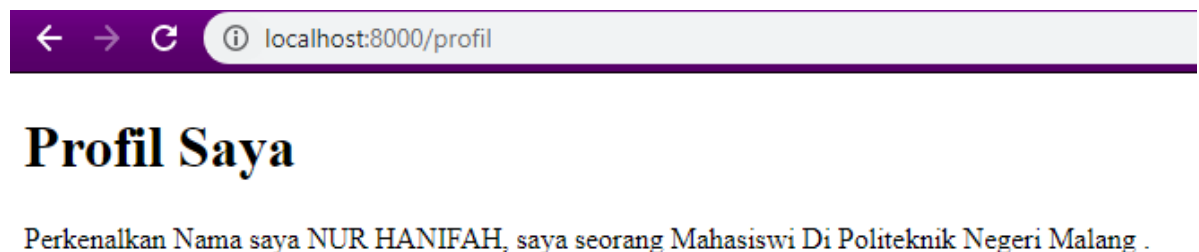
Langkah selanjutnya, tambahkan fungsi / method profil() pada class **CobaController** seperti berikut ini:

```
<?php
namespace App\Http\Controllers;
use Illuminate\Http\Request;

class CobaController extends Controller
{
    public function profil()
    {
        return view('profil');
    }
}
```

Ketikkan alamat <http://localhost:8000/profil>, dan Kita akan mendapatkan hasil yang sama seperti pada latihan sebelumnya.

Hasil:



Praktikum – Bagian 4: Memberikan Data Controller kepada View

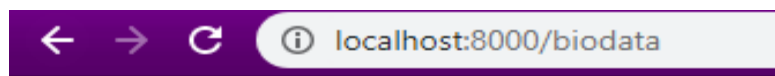
Langkah	Keterangan
1	<p>Dalam sebuah aplikasi, sangat jarang Kita temukan isi dari file view yang bersifat statis. Pada umumnya, view dipakai untuk menampilkan data, contohnya adalah data yang berasal dari database. Namun, karena kita belum sampai pada pembahasan database, maka pada latihan kali ini kita akan mencoba menampilkan data dari variabel.</p> <p>Buat route baru untuk halaman biodata</p> <p style="text-align: center;">Route::get('biodata', 'BiodataController@index');</p> <p>Buat controller baru dengan nama BiodataController.php</p> <p style="text-align: center;">php artisan make:controller BiodataController</p> <p>Hasil:</p> <pre>C:\xampp\htdocs\laravelapp>php artisan make:controller BiodataController Controller created successfully. C:\xampp\htdocs\laravelapp></pre> <p>Ketikkan script berikut dengan memberikan method / fungsi index pada laravelapp\app\Http\Controllers\BiodataController.php</p> <pre><?php namespace App\Http\Controllers; use Illuminate\Http\Request; class BiodataController extends Controller { public function index() { \$nama = 'Sugono Galih Aprianto'; <i>//ubah dengan nama kalian</i> return view('biodata' , ['nama' => \$nama]); } }</pre>

2

Buat file view dengan nama **biodata.blade.php** pada **laravelapp/resources/views**

```
<!DOCTYPE html>
<html>
<head>
    <title>Biodata</title>
</head>
<body>
    <h1>Biodata</h1>
    <p>Nama : {{ $nama }}</p>
</body>
</html>
```

Untuk melihat hasilnya, pada browser, ketik alamat <http://localhost:8000/biodata> .
Hasil:



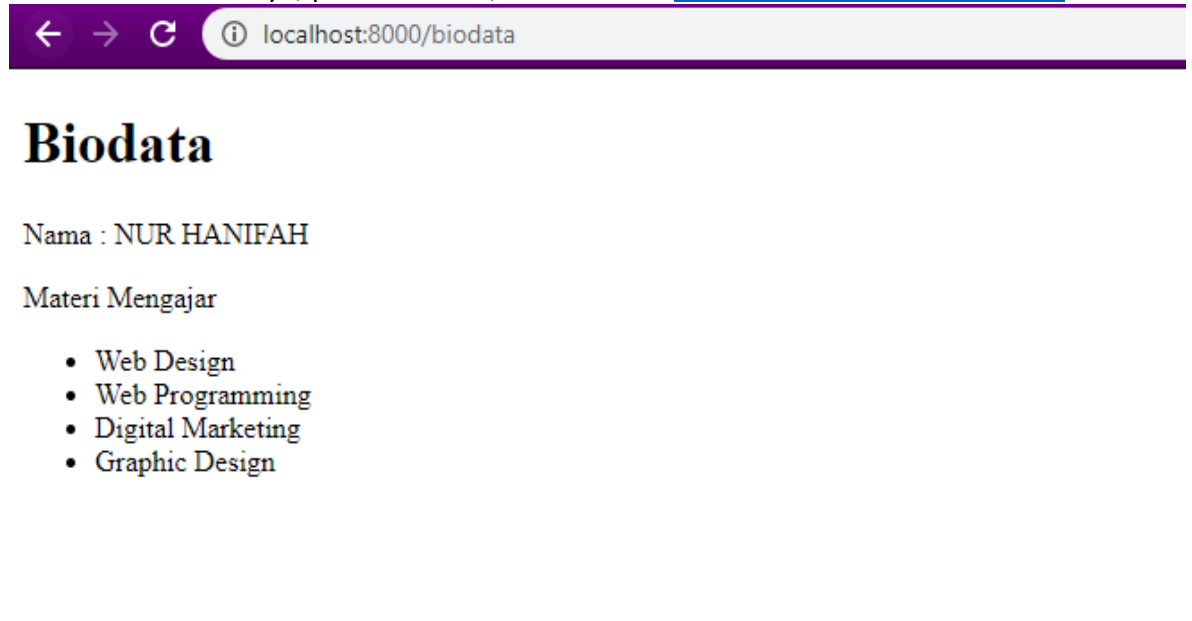
Biodata

Nama : NUR HANIFAH

Praktikum – Bagian 5: Memberikan Data Array Kepada View

Langkah	Keterangan
1	<p>Pada bagian ini, Kita coba untuk memberikan data array kepada view. Tambahkan variabel data array pada BiodataController.php</p> <pre><?php namespace App\Http\Controllers; use Illuminate\Http\Request; class BiodataController extends Controller { public function index() { \$nama = 'Sugono Galih Aprianto'; <i>//ubah dengan nama kalian</i> \$materi = ["Web Design", "Web Programming", "Digital Marketing","Graphic Design"]; return view('biodata' , ['nama' => \$nama, 'materi' => \$materi]); } }</pre>
2	<p>Tambahkan isi file view biodata.blade.php menjadi seperti berikut:</p> <pre><!DOCTYPE html> <html> <head> <title>Biodata</title> </head> <body> <h1>Biodata</h1> <p>Nama : {{ \$nama }}</p> <p>Materi Mengajar</p> @foreach(\$materi as \$datamateri) {{ \$datamateri }} @endforeach </body> </html></pre>

Untuk melihat hasilnya, pada browser, ketik alamat <http://localhost:8000/biodata> .



-- Selamat Mengerjakan --

