

JOBSHEET 9 : PENGANTAR DAN INSTALASI LARAVEL

PEMOGRAMAN WEB LANJUT



POLITEKNIK NEGERI MALANG

APRIL 2020

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> <ol style="list-style-type: none"> 1. Install Via Laravel Installer Setelah proses instalasi Composer selesai, buka Command Prompt lalu tuliskan lokasi folder yang akan menampung folder kerja kita. Karena pada praktikum sebelumnya telah menggunakan XAMPP, maka dapat menggunakan folder htdocs. Pindah ke direktori htdocs dengan ketik: <code>cd C:\xampp\htdocs</code>. Setelah berada di dalam folder, ketikkan perintah sebagai berikut: <code>composer global require "laravel/installer"</code> Tunggulah hingga proses penginstalan selesai. Setelah itu, Kita bisa mulai membuat project laravel dengan mengetikkan perintah: <code>laravel new laravelapp</code> Keterangan: <code>laravelapp</code>: 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: <code>composer create-project --prefer-dist laravel/laravel laravelapp</code>.

	<div><div>OUTPUTTERMINALDEBUG CONSOLEPROBLEMS</div><div><div>1: powershell</div><div><pre>PS C:\Users\AGDS> cd c:\xampp\htdocs PS C:\xampp\htdocs> composer global require "laravel/installer" Updated current directory to C:\Users\AGDS\AppData\Local\Composer Using version ^3.0 for laravel/installer ./composer.json has been created Loading composer repositories with package information Updating dependencies (including require-dev) Package operations: 14 installs, 0 updates, 0 removals - Installing symfony/process (v5.0.7): Downloading (100%) - Installing symfony/polyfill-curl (v1.15.0): Loading from cache - Installing symfony/filesystem (v5.0.7): Downloading (100%) - Installing psr/container (1.0.0): Downloading (100%) - Installing symfony/service-contracts (v2.0.3): Downloading (100%) - Installing symfony/polyfill-php73 (v1.15.0): Downloading (100%) - Installing symfony/polyfill-php80 (v1.15.0): Downloading (100%) - Installing ralouphie/getallheaders (3.0.3): Loading from cache - Installing psr/http-message (1.0.1): Loading from cache - Installing guzzlehttp/psr7 (1.8.1): Loading from cache - Installing guzzlehttp/promises (v1.3.1): Loading from cache - Installing guzzlehttp/guzzle (6.5.2): Loading from cache - Installing laravel/installer (v3.0.3): Downloading (100%) symfony/service-contracts suggests installing symfony/service-implementation symfony/console suggests installing symfony/event-dispatcher symfony/console suggests installing psr/log (for using the console logger) guzzlehttp/psr7 suggests installing zendframework/zend-httphandlerrunner (Edit PSR-7 responses) guzzlehttp/guzzle suggests installing psr/log (Required for using the log middleware) guzzlehttp/guzzle suggests installing ext-intl (Required for Internationalized Domain Name (IDN) support) Writing lock file Generating autoload files 6 packages you are looking for funding. Use the composer fund command to find out more! PS C:\xampp\htdocs> laravel new laravelapp Crafting application...</pre></div></div></div>
3.	<p>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: <code>php artisan serve</code> 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 :</p> <div><div>localhost/laravelapp/public/</div><div><div>Laravel</div><div>DOCS LARACASTS NEWS BLOG NOVA FORGE VAPOR GITHUB</div></div></div> <p>Jalankan dengan localhost:8000</p> <div><div>localhost:8000</div><div><div>Laravel</div><div>DOCS LARACASTS NEWS BLOG NOVA FORGE VAPOR GITHUB</div></div></div>

Praktikum – Bagian 1: Instalasi Laravel di Windows

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> 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 : <code>Route::get('hallo', function () { return "Hallo, Anda sedang mencoba Routing di Laravel"; });</code> Untuk melihat hasilnya, pada browser, ketikkan alamat http://localhost:8000/hallo</p>  <pre> 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('hallo', function () { 17 //return view('welcome'); 18 return "Hallo, Anda Sedang Mencoba Routing di Laravel"; 19 }); </pre> <p>Hallo, Anda Sedang Mencoba Routing di Laravel</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:

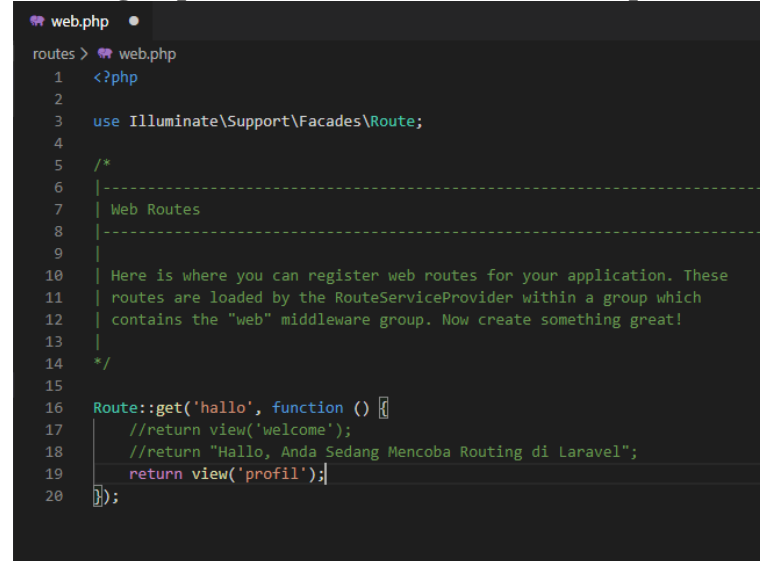
Profil Saya

Perkenalkan Nama saya nama_saya, saya seorang

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

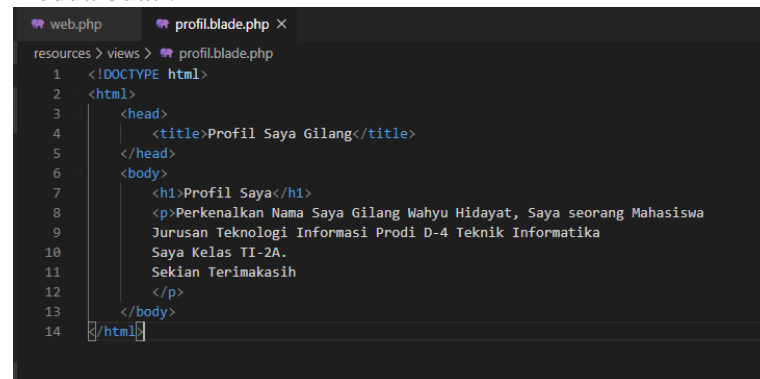
Pertama buat :

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



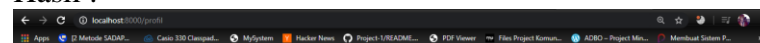
```
web.php
routes > web.php
1  <?php
2
3  use Illuminate\Support\Facades\Route;
4
5  /*
6  |-----
7  | Web Routes
8  |-----
9  |
10 | Here is where you can register web routes for your application. These
11 | routes are loaded by the RouteServiceProvider within a group which
12 | contains the "web" middleware group. Now create something great!
13 |
14 */
15
16 Route::get('hallo', function () {
17     //return view('welcome');
18     //return "Hallo, Anda Sedang Mencoba Routing di Laravel";
19     return view('profil');
20 });
```

Kedua buat :



```
web.php  profil.blade.php x
resources > views > profil.blade.php
1  <!DOCTYPE html>
2  <html>
3      <head>
4          <title>Profil Saya Gilang</title>
5      </head>
6      <body>
7          <h1>Profil Saya</h1>
8          <p>Perkenalkan Nama Saya Gilang Wahyu Hidayat, Saya seorang Mahasiswa
9              Jurusan Teknologi Informasi Prodi D-4 Teknik Informatika
10             Saya Kelas TI-2A.
11             Sekian Terimakasih
12         </p>
13     </body>
14 </html>
```

Hasil :



Profil Saya

Perkenalkan Nama Saya Gilang Wahyu Hidayat, Saya seorang Mahasiswa Jurusan Teknologi Informasi Prodi D-4 Teknik Informatika Saya Kelas TI-2A. Sekian Terimakasih

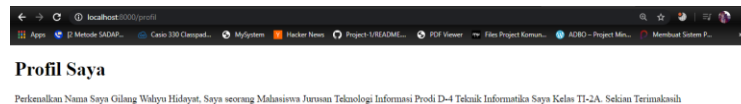
Langkah	Keterangan
1.	<p>Pada pembahasan sebelumnya, Kita sudah bisa menampilkan file view langsung dari Routenya. Tetapi sebuah aplikasi tidaklah sesimpel itu, akan ada logika dan data-data yang harus diolah terlebih dahulu. Caranya adalah dengan menggunakan Controller, seperti yang telah dipelajari ketika menggunakan CodeIgniter. 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. Kita gunakan cara yang kedua, pada latihan kali ini kita akan membuat Controller dengan nama CobaController dengan menuliskan di command prompt / terminal : <code>php artisan make:controller CobaController</code> Maka akan terbentuk sebuah file dengan nama <code>CobaController.php</code> pada folder <code>laravelapp\app\Http\Controllers\</code></p>  <p>Hasil membuat controller :</p> 
2.	<p>Ubah route halaman 'profil' pada praktikum sebelumnya menjadi seperti berikut <code>Route::get('profil', 'CobaController@profil');</code> Langkah selanjutnya, tambahkan fungsi / method <code>profil()</code> pada class <code>CobaController</code> seperti berikut ini: Pertama Ubah :</p>

```
CobaController.php  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('profil','CobaController@profil');
17 //return view('welcome');
18 //return "Hallo, Anda Sedang Mencoba Routing di Laravel";
19 //return view('profil');
20 //});
```

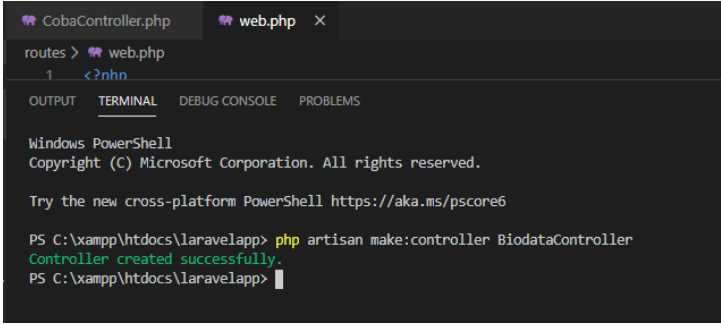
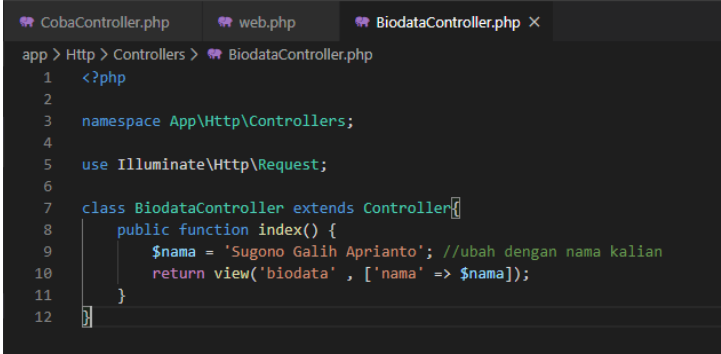
Kedua Ubah :

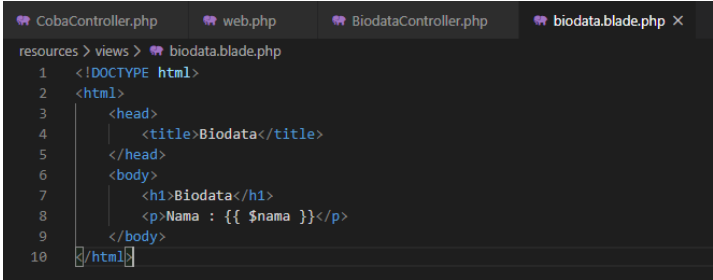
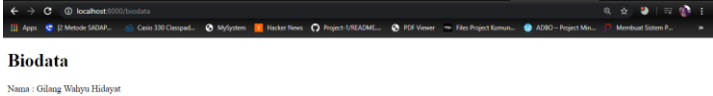
```
CobaController.php X  web.php
app > Http > Controllers > CobaController.php
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6
7  class CobaController extends Controller{
8
9      public function profil(){
10         return view('profil');
11     }
12 }
13
```

Hasil :

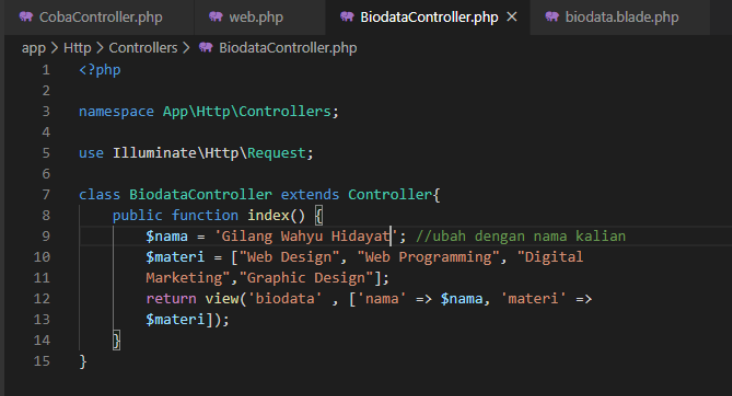
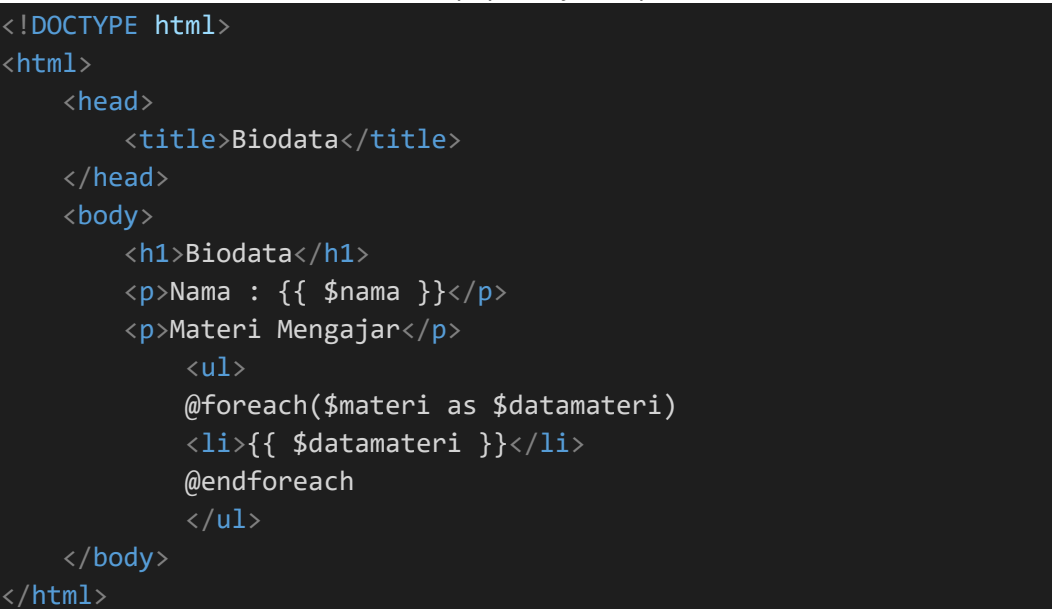
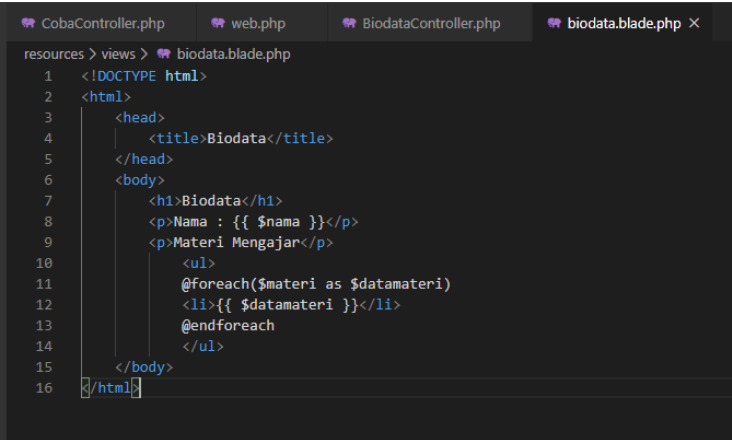


Praktikum – Bagian 3: Cara Menggunakan Controller pada Framework Laravel

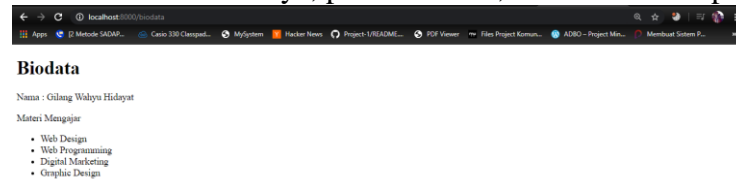
Langkah	Keterangan
1.	<p>Dalam sebuah aplikasi, sangat jarang Kita temukan isi dari file view yang bersifat statis. Pada umumnya, view dipakai untuk menampilkan data, contohnya adalah data yang berasal dari database. Namun, karena kita belum sampai pada pembahasan database, maka pada latihan kali ini kita akan mencoba menampilkan data dari variabel. Buat route baru untuk halaman biodata <code>Route::get('biodata', 'BiodataController@index');</code> Buat controller baru dengan nama <code>BiodataController.php</code> php artisan make:controller BiodataController Ketikkan script berikut dengan memberikan method / fungsi index pada <code>laravelapp\app\Http\Controllers\BiodataController.php \$nama]); } }</code></p> <p>Tahap Pertama :</p>  <p>Tahap Kedua :</p>  <p>Tahap Ketiga :</p> 

2.	<p>Buat file view dengan nama biodata.blade.php pada laravelapp\resources\views\</p> <p>Biodata</p> <p>Nama : {{ \$nama }}</p> <p>Untuk melihat hasilnya, pada browser, ketik alamat http://localhost:8000/biodata .</p> <p>Tahap Pertama :</p>  <pre> resources > views > biodata.blade.php 1 <!DOCTYPE html> 2 <html> 3 <head> 4 <title>Biodata</title> 5 </head> 6 <body> 7 <h1>Biodata</h1> 8 <p>Nama : {{ \$nama }}</p> 9 </body> 10 </html> </pre> <p>Hasil :</p>  <p>Biodata</p> <p>Nama : Gilang Wahyu Hidayat</p>
----	---

Praktikum – Bagian 4: Memberikan Data Controller kepada View

Langkah	Keterangan
1.	<p>Pada bagian ini, Kita coba untuk memberikan data array kepada view. Tambahkan variabel data array pada BiodataController.php \$nama, 'materi' => \$materi]); } }</p>  <pre> 1 <?php 2 3 namespace App\Http\Controllers; 4 5 use Illuminate\Http\Request; 6 7 class BiodataController extends Controller{ 8 public function index() { 9 \$nama = 'Gilang Wahyu Hidayat'; //ubah dengan nama kalian 10 \$materi = ["Web Design", "Web Programming", "Digital 11 Marketing", "Graphic Design"]; 12 return view('biodata', ['nama' => \$nama, 'materi' => 13 \$materi]); 14 } 15 } </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>  <pre> 1 <!DOCTYPE html> 2 <html> 3 <head> 4 <title>Biodata</title> 5 </head> 6 <body> 7 <h1>Biodata</h1> 8 <p>Nama : {{ \$nama }}</p> 9 <p>Materi Mengajar</p> 10 11 @foreach(\$materi as \$datamateri) 12 {{ \$datamateri }} 13 @endforeach 14 15 </body> 16 </html> </pre>

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



Praktikum – Bagian 5: Memberikan Data Array Kepada View