



Mata Kuliah : Pemrograman Web Lanjut (PWL)
Program Studi : D4 – Sistem Informasi Bisnis
Semester : 5

Kelas : SIB 3D
NIM : 2241760030
Nama : Fardiyani Afro'ul Yasinta
Jobsheet Ke- : 2

Laporan Jobsheet

Praktikum Ke-1

- Basic Routing

Langkah	Jawaban/Deskripsi												
1.	<p>Pada bagian ini, kita akan membuat dua buah route dengan ketentuan sebagai berikut.</p> <table><tr><th>No</th><th>Http Verb</th><th>Url</th><th>Fungsi</th></tr><tr><td>1</td><td>get</td><td>/hello</td><td>Tampilkan String Hello ke browser.</td></tr><tr><td>2</td><td>get</td><td>/world</td><td>Tampilkan String World ke browser</td></tr></table>	No	Http Verb	Url	Fungsi	1	get	/hello	Tampilkan String Hello ke browser.	2	get	/world	Tampilkan String World ke browser
No	Http Verb	Url	Fungsi										
1	get	/hello	Tampilkan String Hello ke browser.										
2	get	/world	Tampilkan String World ke browser										
2	<p>Buka file <code>routes/web.php</code>. Tambahkan sebuah route untuk nomor 1 seperti di bawah ini:</p> <pre>use Illuminate\Support\Facades\Route; Route::get('/hello', function () { return 'Hello World'; });</pre>												
3	<p>Buka browser, tuliskan URL untuk memanggil route tersebut: <code>localhost/PWL_2024/public/hello</code>. Perhatikan halaman yang muncul apakah sudah sesuai dan jelaskan pengamatan Anda.</p>												



	<pre>Route::get('/hello', function () { return ('Hello World'); });</pre> <p>kode diatas mendefinisikan route /hello yang ketika diakses, akan menampilkan teks "Hello World" di halaman web.</p> <p>Output</p> <hr/> <p>Hello World</p>
4.	<p>Untuk membuat route kedua, tambahkan route /world seperti di bawah ini:</p> <pre>use Illuminate\Support\Facades\Route; Route::get('/world', function () { return 'World'; });</pre>
5.	<p>Bukalah pada browser, tuliskan URL untuk memanggil route tersebut: localhost/PWL_2024/public/world. Perhatikan halaman yang muncul apakah sudah sesuai dan jelaskan pengamatan Anda.</p> <pre>Route::get('/world', function () { return ('World'); });</pre> <p>kode diatas mendefinisikan route /world yang ketika diakses, akan menampilkan teks "World" di halaman web.</p> <p>Output</p>



	<p>World</p>
6	<p>Selanjutnya, cobalah membuat route '/' yang menampilkan pesan 'Selamat Datang'.</p> <pre>Route::get('/', function () { return ('Selamat Datang '); });</pre> <p>kode diatas mendefinisikan route / yang ketika diakses, akan menampilkan teks "Selamat Datang" di halaman web.</p> <p>Output</p> <p>Selamat Datang</p>
7	<p>Kemudian buatlah route '/about' yang akan menampilkan NIM dan nama Anda.</p> <pre>Route::get('/about', function () { return ('Fardiyani Afroul Yasinta 2241760030'); });</pre> <p>kode diatas mendefinisikan route /about yang ketika diakses, akan menampilkan teks "Fardiyani Afroul Yasinta 2241760030" di halaman web.</p> <p>Output</p> <p>Fardiyani Afroul Yasinta 2241760030</p>
8.	<p>Kita akan memanggil route /user/{name} sekaligus mengirimkan parameter berupa nama user \$name seperti kode di bawah ini.</p>

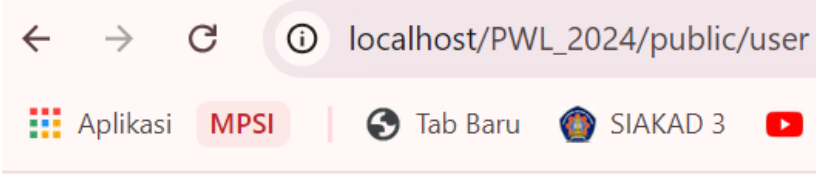


	<pre>Route::get('/user/{name}', function (\$name) { return 'Nama saya '.\$name; });</pre>	
9.	<p>Jalankan kode dengan menuliskan URL untuk memanggil route tersebut: localhost/PWL_2024/public/user>NamaAnda. Perhatikan halaman yang muncul dan jelaskan pengamatan Anda.</p> <pre>Route::get('/user/{name}', function (\$name) { return 'Nama Saya '.\$name; });</pre> <p>Output localhost/PWL_2024/public/user/Fardiyani Afro'ul Yasinta</p> <p>Nama Saya Fardiyani Afro'ul Yasinta</p> <p>Pada file router terdapat perintah untuk membuat url /user/ Fardiyani Afro'ul Yasinta dan mereturn string Nama saya dan memanggil parameter \$nama, jadi pada saat membuka url localhost/PWL_2024/public/user/ Fardiyani Afro'ul Yasinta maka akan menampilkan tulisan Nama saya Fardiyani Afro'ul Yasinta</p>	
10.	<p>Selanjutnya, coba tuliskan URL: localhost/PWL_2024/public/user/. Perhatikan halaman yang muncul dan jelaskan pengamatan Anda</p> <p>Output localhost/PWL_2024/public/user/</p>	



	 <p>Pada saat browser dijalankan akan muncul 404 not found karena tidak ada url yang mengarah ke /user.</p>
11.	<p>Suatu route, juga bisa menerima lebih dari 1 parameter seperti kode berikut ini. Route menerima parameter \$postId dan juga \$comment.</p> <pre>Route::get('/posts/{post}/comments/{comment}', function (\$postId, \$commentId) { return 'Pos ke-' . \$postId . " Komentar ke-: " . \$commentId; });</pre>
12.	<p>Jalankan kode dengan menuliskan URL untuk memanggil route tersebut: localhost/PWL_2024/public/posts/1/comments/5. Perhatikan halaman yang muncul dan jelaskan pengamatan Anda.</p> <pre>Route::get('/posts/{post}/comments/{comment}', function (\$postId, \$commentId) { return 'Pos ke- ' . \$postId . "Komentar ke- " . \$commentId; });</pre> <p>Output</p>  <p>Fungsi tersebut kemudian akan mengembalikan string yang terdiri dari teks "Pos ke-1 Komentar ke-5", sesuai dengan nilai dari \$postId dan \$commentId</p>



13.	<p>Kemudian buatlah route <code>/articles/{id}</code> yang akan menampilkan output “Halaman Artikel dengan ID {id}”, ganti id sesuai dengan input dari url.</p> <pre>Route::get('/articles/{id}', function (\$id) { return 'Halaman Artikel dengan ID '.\$id; });</pre> <p>Output</p> <p>Halaman Artikel dengan ID 150</p>
14.	<p>Kita akan memanggil route <code>/user</code> sekaligus mengirimkan parameter berupa nama user <code>\$name</code> dimana parameternya bersifat opsional.</p> <pre>Route::get('/user/{name?}', return 'Nama saya '.\$name; });</pre>
15.	<p>Jalankan kode dengan menuliskan URL: localhost/PWL_2024/public/user/. Perhatikan halaman yang muncul dan jelaskan pengamatan Anda.</p> <pre>Route::get('/user/{name?}', function (\$name=null) { return 'Nama Saya '.\$name; });</pre> <p>Output</p> <p>localhost/PWL_2024/public/user/</p>  <p>Nama Saya</p> <p>Pada fungsi name saya mengisi parameter dengan null sehingga apabila url <code>/user</code> dipanggil di browse maka akan menampilkan nama saya</p>



16.	<p>Selanjutnya tuliskan URL: localhost/PWL_2024/public/user>NamaAnda. Perhatikan halaman yang muncul dan jelaskan pengamatan Anda</p> <p>Jawab:</p> <p>localhost/PWL_2024/public/user/Fardiyani Afro'ul Yasinta</p>  <p>Nama Saya Fardiyani Afro'ul Yasinta</p> <p>Pada percobaan kali ini saya mengganti parameter name='null' menjadi Fardiyani Afro'ul Yasinta. Yang membedakan percobaan ini dengan percobaan sebelumnya dimana pemanggilan url /user/ Fardiyani Afro'ul Yasinta akan menampilkan Fardiyani Afro'ul Yasinta karena parameter \$name itu kosong dan tidak ada nilainya, sehingga saat memanggil url /user saja akan menampilkan 404 not found. Sedangkan pada percobaan kali ini route parameternya name terdapat operator tanya (?) dimana nilai dari name harus di definisikan terlebih dahulu sehingga apabila kita memanggil url /user saja, browser akan langsung menampilkan 'Nama saya Fardiyani Afro'ul Yasinta'.</p>
17.	<p>Ubah kode pada route /user menjadi seperti di bawah ini.</p> <pre>Route::get('/user/{name?}', function (\$name='John') { return 'Nama saya '.\$name; });</pre>
18.	<p>Jalankan kode dengan menuliskan URL: localhost/PWL_2024/public/user/. Perhatikan halaman yang muncul dan jelaskan pengamatan Anda.</p>  <p>Output</p>



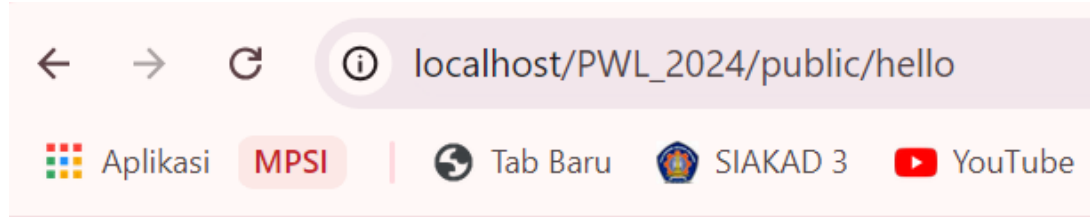
--	--

Praktikum Ke-2

Langkah	Jawaban/Deskripsi
1.	<p>Untuk membuat controller pada Laravel telah disediakan perintah untuk menggenerate struktur dasarnya. Kita dapat menggunakan perintah artisan diikuti dengan definisi nama controller yang akan dibuat.</p> <p>php artisan make:controller WelcomeController</p>
2.	<p>Buka file pada app/Http/Controllers/WelcomeController.php. Struktur pada controller dapat digambarkan sebagai berikut:</p> <pre><?php namespace App\Http\Controllers; use Illuminate\Http\Request; class WelcomeController extends Controller { // }</pre>
3.	<p>Untuk mendefinisikan action, silahkan tambahkan function dengan access public. Sehingga controller di atas menjadi sebagai berikut:</p>



	<pre><?php namespace App\Http\Controllers; use Illuminate\Http\Request; class WelcomeController extends Controller { public function hello() { return 'Hello World'; } }</pre>	
4.	Setelah sebuah controller telah didefinisikan action, kita dapat menambahkan controller tersebut pada route. Ubah route /hello menjadi seperti berikut: Route::get('/hello', [WelcomeController::class,'hello']);	
5.	<p>Buka browser, tuliskan URL untuk memanggil route tersebut: localhost/PWL_2024/public/hello. Perhatikan halaman yang muncul dan jelaskan pengamatan Anda.</p> <pre>app > Http > Controllers > WelcomeController.php > ... 1 <?php 2 3 namespace App\Http\Controllers; 4 5 use Illuminate\Http\Request; 6 7 2 references 0 implementations 8 class WelcomeController extends Controller 9 { 10 1 reference 0 overrides 11 public function hello() { 12 return 'Hello World'; 13 } 14 }</pre> <p>Output</p>	



Hello World

Pada percobaan kali ini saya menggunakan file baru bernama WelcomeController sebagai induk filenya dan pada file web.php dilakukan pemanggilan controller untuk menampilkan 'hello' saat url /hello dipanggil di browser.

6

Modifikasi hasil pada praktikum poin 2 (Routing) dengan konsep controller. Pindahkan logika eksekusi ke dalam controller dengan nama PageController.

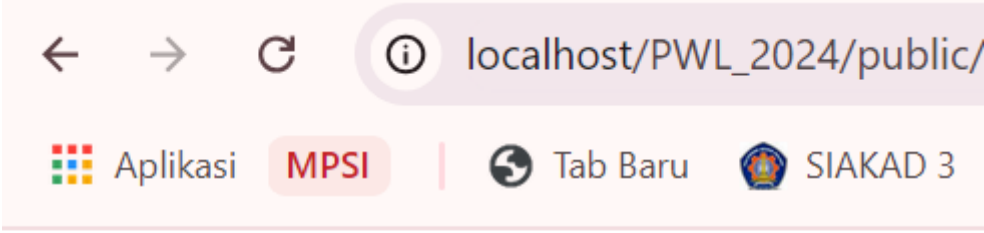
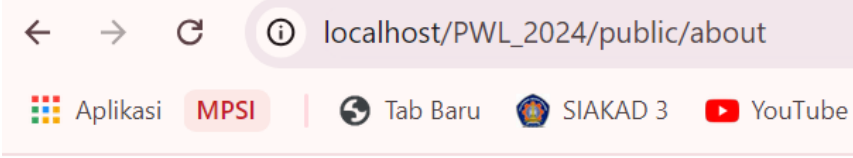
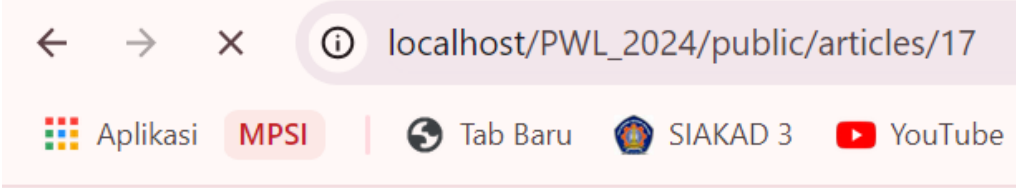
Resource	POST	GET	PUT	DELETE
/		Tampilkan Pesan 'Selamat Datang' PageController : index		
/about		Tampilkan Nama dan NIM PageController : about		
/articles/		Tampilkan halaman dinamis		

{id}		'Halaman Artikel dengan Id {id}' id diganti sesuai input dari url PageController : articles		
------	--	--	--	--

PageController.php


```
app > Http > Controllers > PageController.php > PHP Intelephense > PageController > about
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6
7  4 references | 0 implementations
8  class PageController extends Controller
9  {
10     1 reference | 0 overrides
11     public function index() {
12         return 'Selamat Datang';
13     }
14     1 reference | 0 overrides
15     public function about() {
16         return 'Nama: Fardiyani Afroul Yasinta <br> NIM: 2241760030';
17     }
18     1 reference | 0 overrides
19     public function articles($id) {
20         return 'Halaman Artikel dengan ID '. $id;
21     }
22 }
```



	<p>Web.php</p> <pre>Route::get('/', [PageController::class, 'index']);</pre> <p>Output</p>  <p>Selamat Datang</p> <pre>Route::get('/about', [PageController::class, 'about']);</pre> <p>Output</p>  <p>Nama: Fardiyani Afroul Yasinta NIM: 2241760030</p> <pre>Route::get('/articles/{id}', [PageController::class, 'articles']);</pre> <p>Output</p>  <p>Halaman Artikel dengan ID 17</p>
7	Modifikasi kembali implementasi sebelumnya dengan konsep Single Action Controller. Sehingga untuk hasil akhir yang didapatkan akan ada HomeController, AboutController dan ArticleController. Modifikasi juga route yang digunakan.

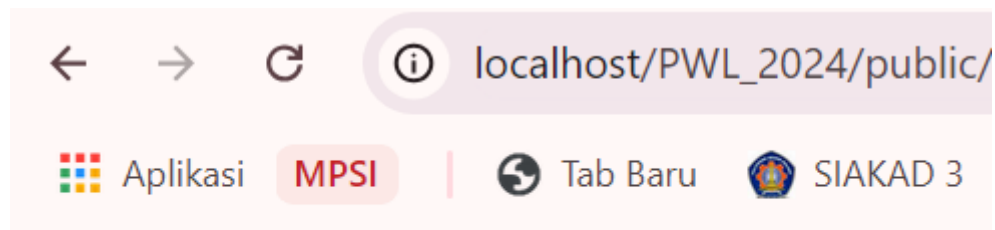


Home Controller

```
app > Http > Controllers >  HomeController.php > ...  
1  <?php  
2  
3  namespace App\Http\Controllers;  
4  
5  use Illuminate\Http\Request;  
6  
7  2 references | 0 implementations  
8  class HomeController extends Controller  
9  {  
10     1 reference | 0 overrides  
11     public function index() {  
12         return 'Selamat Datang';  
13     }  
14 }
```

```
Route::get('/', [HomeController::class, 'index']);
```

Output



Selamat Datang

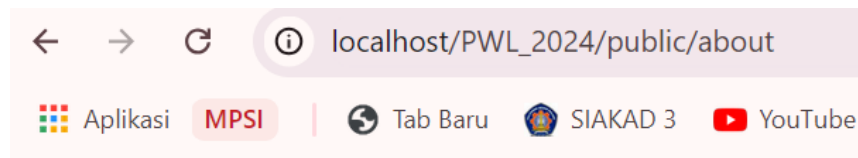


About Controller

```
app > Http > Controllers > AboutController.php > ...
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6
7  2 references | 0 implementations
8  class AboutController extends Controller
9  {
10     1 reference | 0 overrides
11     public function about() {
12         return 'Nama: Fardiyani Afroul Yasinta <br> NIM: 2241760030';
13     }
14 }
```

```
Route::get('/about', [AboutController::class, 'about']);
```

Output



Nama: Fardiyani Afroul Yasinta
NIM: 2241760030

Article Controller

```
app > Http > Controllers > ArticleController.php > ...
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6
7  2 references | 0 implementations
8  class ArticleController extends Controller
9  {
10     1 reference | 0 overrides
11     public function articles($id) {
12         return 'Halaman Artikel dengan ID '. $id;
13     }
14 }
```





	<pre>GET HEAD photos photos.index > PhotoController@index POST photos photos.store > PhotoController@store GET HEAD photos/create photos.create > PhotoController@create GET HEAD photos/{photo} photos.show > PhotoController@show PUT PATCH photos/{photo} photos.update > PhotoController@update DELETE photos/{photo} photos.destroy > PhotoController@destroy GET HEAD photos/{photo}/edit photos.edit > PhotoController@edit GET HEAD posts/{post}/comments/{comment} GET HEAD sanctum/csrf-cookie sanctum.csrf-cookie > Laravel\Sanctum > Cs... GET HEAD user/{name?} GET HEAD user/{name} GET HEAD world</pre>
--	---

Praktikum Ke-3

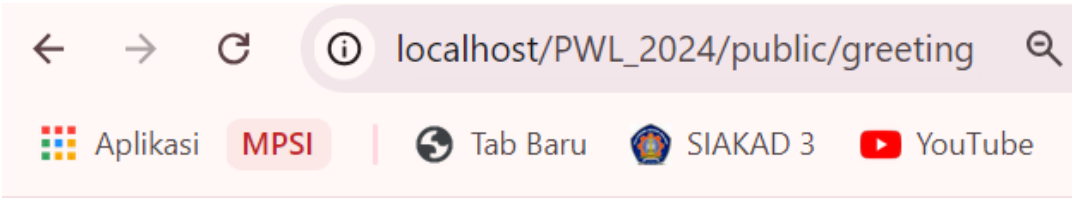
- View

Langkah	Jawaban/Deskripsi
1.	<p>Pada direktori app/resources/views, buatlah file hello.blade.php.</p> <pre><!-- View pada resources/views/hello.blade.php --> <html> <body> <h1>Hello, {{ \$name }}</h1> </body> </html></pre> <pre>resources > views > hello.blade.php > ... 1 <!-- View pada resources/views/hello.blade.php --> 2 <html> 3 <body> 4 <h1>Hello, {{ \$name }}</h1> 5 </body> 6 </html> 7</pre>
2.	<p>View tersebut dapat dijalankan melalui Routing, dimana route akan memanggil View sesuai dengan nama file tanpa 'blade.php'. (Catatan: Gantilah Andi dengan nama Anda)</p> <pre>Route::get('/greeting', function () { return view('hello', ['name' => 'Andi']); });</pre>



3.	<p>Jalankan code dengan membuka url localhost/PWL_2024/public/greeting. Perhatikan halaman yang muncul dan jelaskan pengamatan Anda.</p> <pre>Route::get('/greeting', function () { return view('hello', ['name' => 'Fardiyani Afroul Yasinta']); });</pre> <p>Output</p>  <p>Hello, Fardiyani Afroul Yasinta</p> <p>Output "Hello, Fardiyani Afro'ul Yasinta" muncul karena nilai `name` diisi dengan "Fardiyani Afro'ul Yasinta" dalam route, yang kemudian digunakan dalam template `hello.blade.php` untuk menampilkan pesan tersebut di halaman web.</p>
4.	Buatlah direktori blog di dalam direktori views.
5.	<p>Pindahkan file hello.blade.php ke dalam direktori blog.</p> 
6.	<p>Selanjutnya lakukan perubahan pada route.</p> <pre>Route::get('/greeting', function () { return view('blog.hello', ['name' => 'Andi']); });</pre>
7.	<p>Jalankan code dengan membuka url localhost/PWL_2024/public/greeting. Perhatikan halaman yang muncul dan jelaskan pengamatan Anda.</p> <pre>Route::get('/greeting', function () { return view('blog.hello', ['name' => 'Fardiyani Afroul Yasinta']); });</pre>



	<p>Output</p>  <p>Hello, Fardiyani Afroul Yasinta</p> <p>Route yang telah diubah memanggil view blog.hello, yaitu file hello.blade.php yang telah dipindahkan ke dalam direktori blog. Karena data 'name' diberikan dengan nilai Fardiyani Afro'ul Yasinta, teks Hello, {{ \$name }} di template hello.blade.php akan digantikan dengan "Hello, Fardiyani Afro'ul Yasinta ". Output inilah yang kemudian ditampilkan di halaman web.</p>
8.	<p>Buka WelcomeController.php dan tambahkan fungsi baru yaitu greeting.</p> <pre>class WelcomeController extends Controller { public function hello(){ return('Hello World'); } public function greeting(){ return view('blog.hello', ['name' => 'Andi']); } }</pre>
9.	<p>Ubah route /greeting dan arahkan ke WelcomeController pada fungsi greeting.</p> <pre>Route::get('/greeting', [WelcomeController::class, 'greeting']);</pre>
10.	<p>Jalankan code dengan membuka url localhost/PWL_2024/public/greeting. Perhatikan halaman yang muncul dan jelaskan pengamatan Anda.</p>

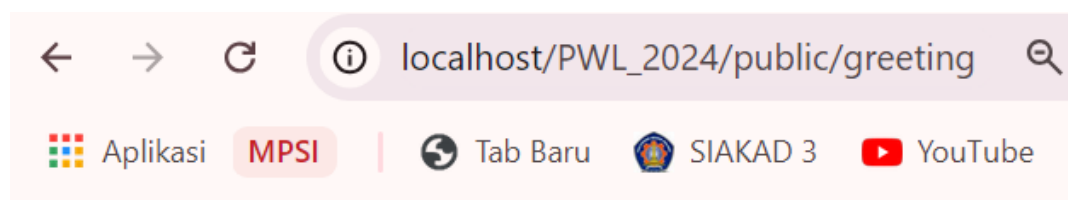


3 references | 0 implementations

```
class WelcomeController extends Controller
{
    1 reference | 0 overrides
    public function hello() {
        return 'Hello World';
    }
    1 reference | 0 overrides
    public function greeting(){
        return view('blog.hello', ['name' => 'Fardiyani Afroul Yasinta']);
    }
}
```

```
Route::get('/greeting', [WelcomeController::class, 'greeting']);
```

Output



Hello, Fardiyani Afroul Yasinta

Padaa folder View blog.hello menerima nilai 'name' sebagai 'Nasywa Syafinka W', dan file hello.blade.php menggantikan teks {{ \$name }} dengan "Nasywa Syafinka W", sehingga output yang muncul adalah "Hello, Nasywa Syafinka W".

11. Buka WelcomeController.php dan tambahkan ubah fungsi greeting.

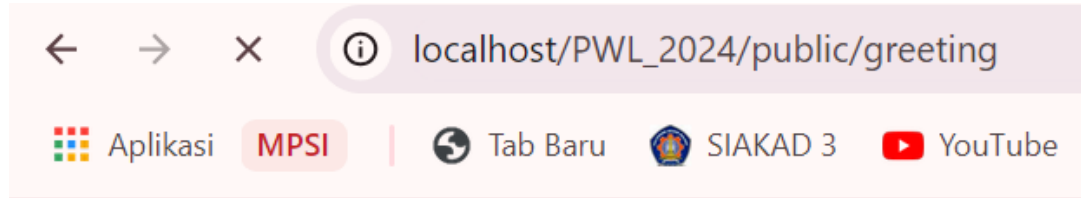
```
class WelcomeController extends Controller
```



	<pre>{ public function hello(){ return('Hello World'); } public function greeting(){ return view('blog.hello') ->with('name', 'Andi') ->with('occupation', 'Astronaut'); } }</pre>
12.	<p>Ubah hello.blade.php agar dapat menampilkan dua parameter.</p> <pre><html> <body> <h1>Hello, {{ \$name }}</h1> <h1>You are {{ \$occupation }}</h1> </body> </html></pre>
13.	<p>Jalankan code dengan membuka url localhost/PWL_2024/public/greeting. Perhatikan halaman yang muncul dan jelaskan pengamatan Anda.</p> <p>Wellcome Controller</p> <pre>public function greeting(){ return view('blog.hello') ->with('name', 'Fardiyani Afroul Yasinta') ->with('occupation', 'Software Engineering'); }</pre> <p>Hello.blade.php</p> <pre><html> <body> <h1>Hello, {{ \$name }}</h1> <h1>You are {{ \$occupation }}</h1> </body> </html></pre>



Output

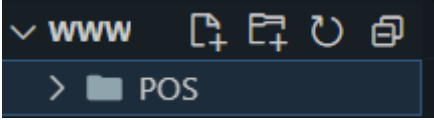


Hello, Fardiyani Afroul Yasinta

You are Software Engineering

Percobaan terakhir ini menggunakan Request dikirim ke route yang mengarah ke method greeting() di WelcomeController. Controller memproses request dan menyiapkan data ('name' dan 'occupation'). Data dikirim ke view (greeting.blade.php) lalu folder View akan menampilkan di browser, menunjukkan nama dan pekerjaan sesuai dengan data yang dikirim dari controller.

Soal Praktikum

Langkah	Jawaban/Deskripsi
1	Buatlah project baru dengan nama POS. Project ini merupakan sebuah aplikasi Point of Sales yang digunakan untuk membantu penjualan. 
2	Buatlah project baru dengan nama POS. Project ini merupakan sebuah aplikasi Point of Sales yang digunakan untuk membantu penjualan.
3	Buatlah beberapa route, controller, dan view sesuai dengan ketentuan sebagai berikut. Route



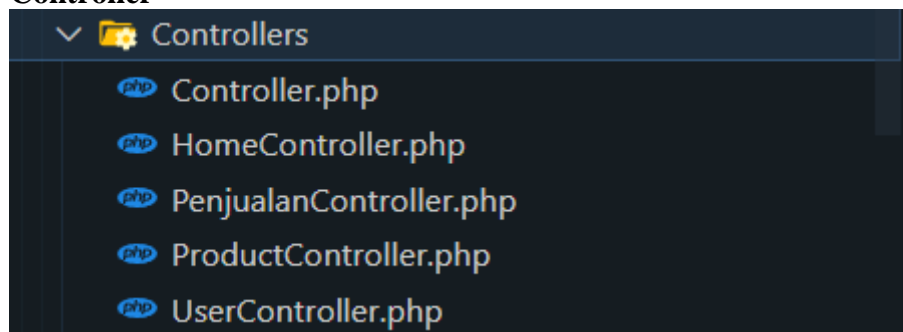
```
// home
Route::get('/', [HomeController::class, 'index']);

// produk
Route::prefix('category')->group(function () {
    Route::get('food-beverage', [ProductController::class, 'foodBeverage']);
    Route::get('beauty-health', [ProductController::class, 'beautyHealth']);
    Route::get('home-care', [ProductController::class, 'homeCare']);
    Route::get('baby-kid', [ProductController::class, 'babyKid']);
});

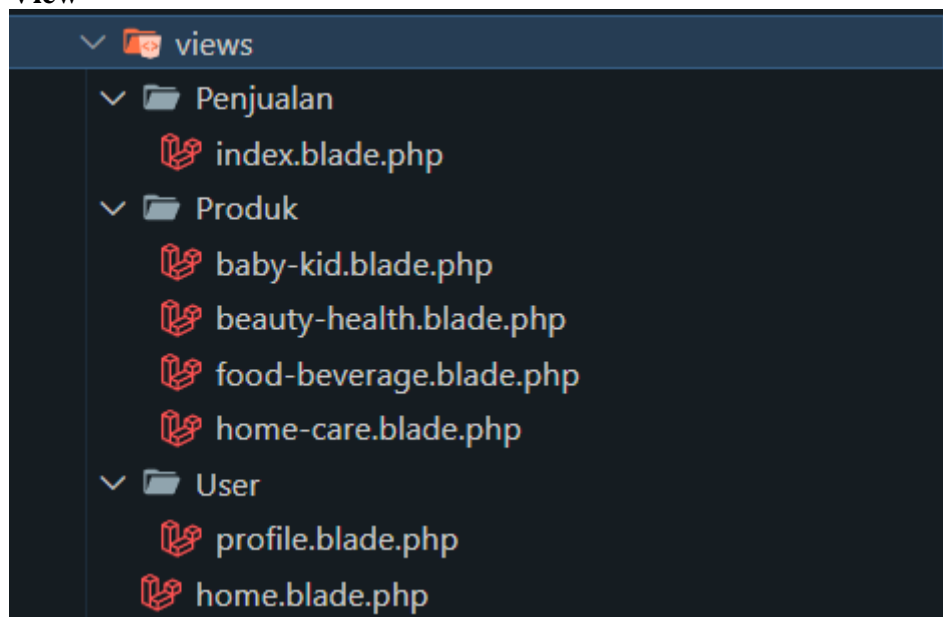
// user
Route::get('/user/{id}/name/{name}', [UserController::class, 'showProfile']);

// penjualan
Route::get('/penjualan', [PenjualanController::class, 'index']);
```

Controller



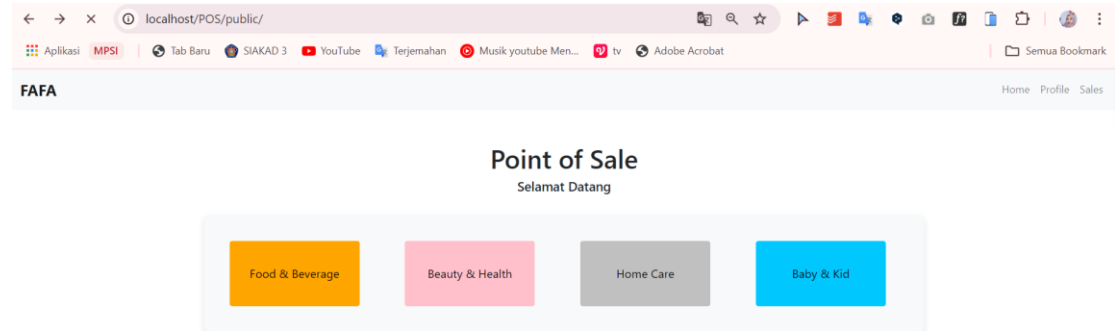
View



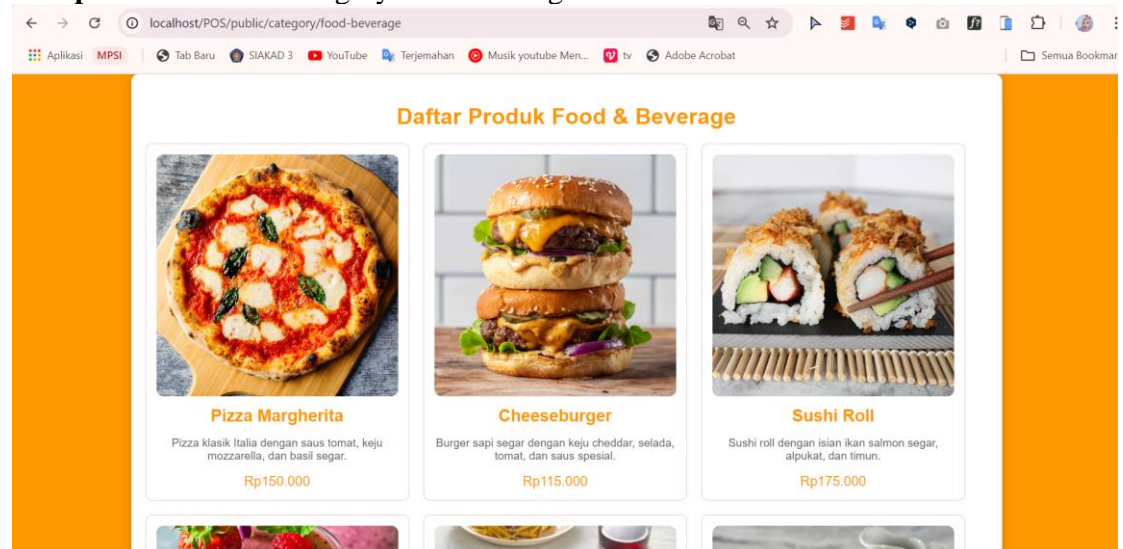


Halaman Home
Menampilkan halaman awal website

Jawab:

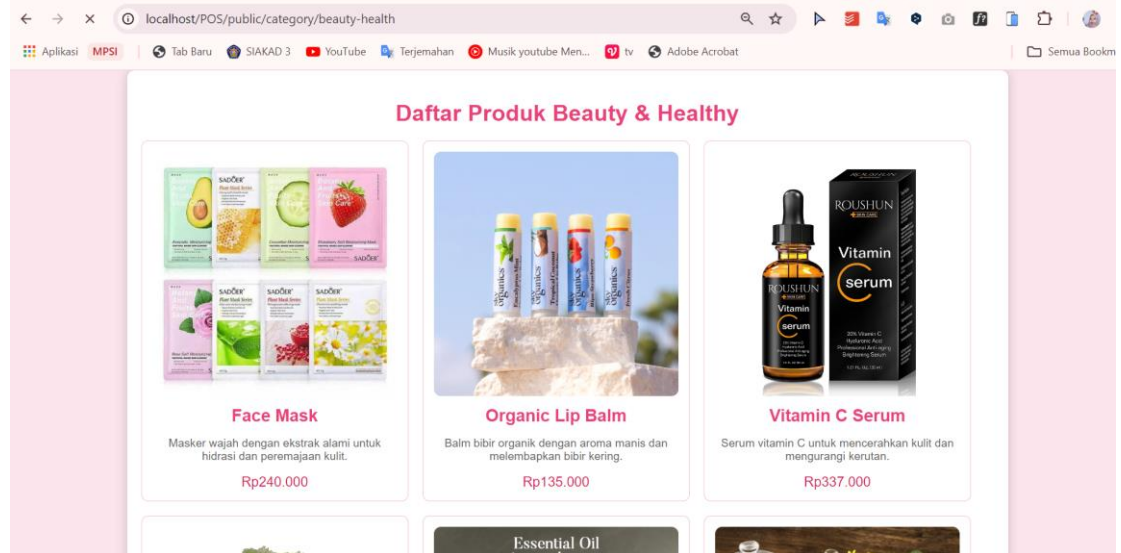


4.
Halaman Products
Menampilkan daftar product (route prefix)
/category/food-beverage
/category/beauty-health
/category/home-care
/category/baby-kid
Jawab:
Tampilan Halaman /category/food-beverage

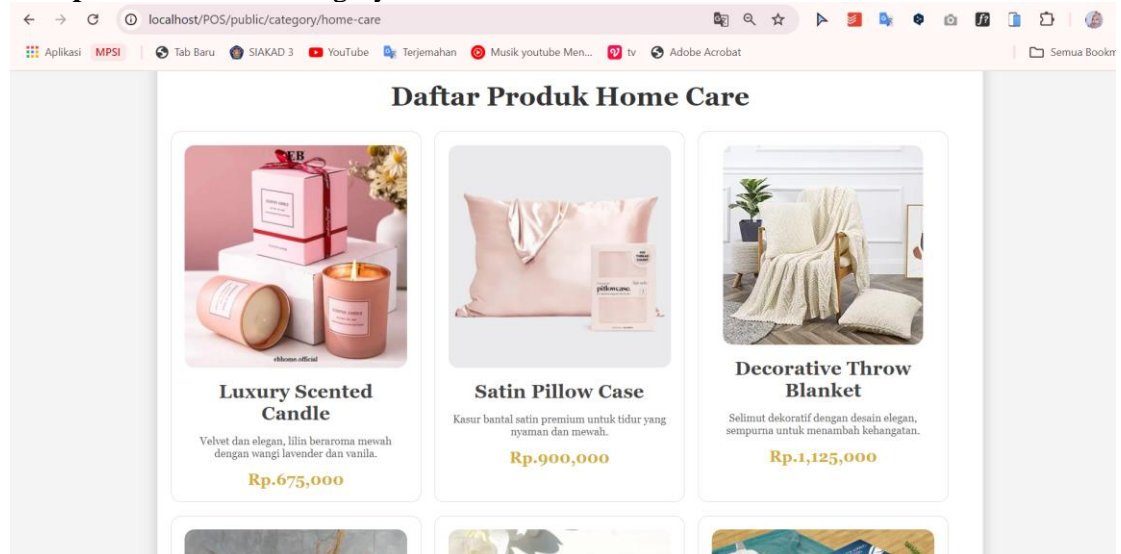




Tampilan Halaman /category/beauty-health



Tampilan Halaman /category/home-care



Tampilan Halaman /category/baby-kid



4.	<p>Halaman User Menampilkan profil pengguna (route param) <code>/user/{id}/name/{name}</code> Jawab:</p>
5.	<p>Halaman Penjualan Menampilkan halaman transaksi POS Jawab:</p>