


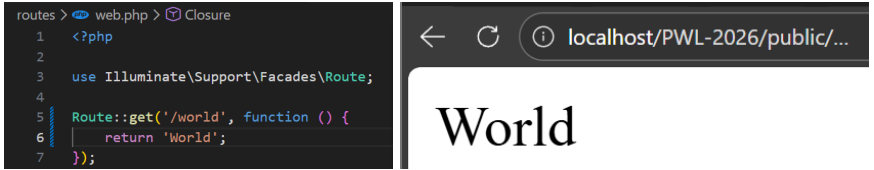
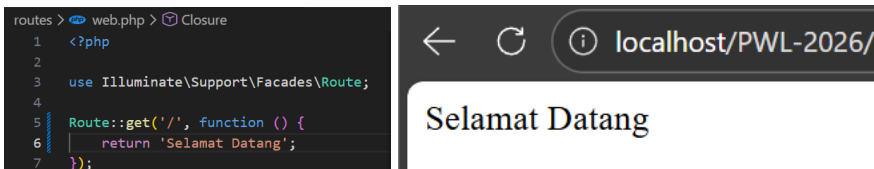


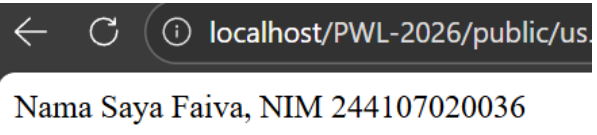
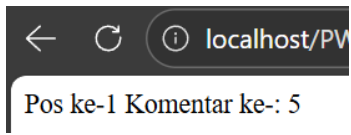
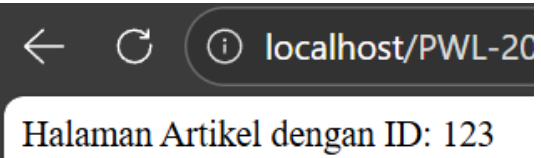
Mata Kuliah : Pemrograman Web Lanjut (PWL)
Program Studi : D-IV Teknik Informatika
Semester : 4

Kelas : TI 2H
NIM : 244107020036
Nama : Faiva Puspa Sahara
Jobsheet Ke- : 2

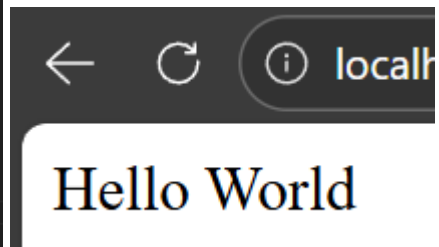
Laporan Jobsheet

Praktikum Ke-1

Langkah	Jawaban/Deskripsi
1	
2	
3	
4	
<i>Basic routing</i> untuk menampilkan teks string ke browser, mencetak "Hello World", "Selamat Datang", dan data Nama beserta NIM.	

5	<pre> routes > web.php > Closure 1 <?php 2 3 use Illuminate\Support\Facades\Route; 4 5 Route::get('/user/{Faiva}', function (\$Faiva) { 6 return "Nama Saya \$Faiva, NIM 244107020036"; 7 }); </pre>  <p>Nama Saya Faiva, NIM 244107020036</p> <p>=> Mempraktikkan Route Parameters tunggal untuk meneruskan satu variabel dinamis dari URL</p>
6	<pre> routes > web.php > ... 1 <?php 2 3 use Illuminate\Support\Facades\Route; 4 5 Route::get('/posts/{post}/comments/{comment}', function (\$postId, \$commentId) { 6 return "Pos ke-".\$postId." Komentar ke-: ".\$commentId; 7 }); </pre>  <p>Pos ke-1 Komentar ke-: 5</p> <p>=> Mempraktikkan Multiple Route Parameters untuk menangkap dan menampilkan lebih dari satu variabel dari URL</p>
7	<pre> routes > web.php > Closure 1 <?php 2 3 use Illuminate\Support\Facades\Route; 4 5 Route::get('/articles/{id}', function (\$id) { 6 return 'Halaman Artikel dengan ID: ' . \$id; 7 }); </pre>  <p>Halaman Artikel dengan ID: 123</p> <p>=> Mempraktikkan Optional Parameters menggunakan tanda tanya (?) agar parameter pada URL bersifat opsional</p>

Praktikum Ke-2

Langkah	Jawaban/Deskripsi
1	<pre> app > Http > Controllers > WelcomeController.php > WelcomeController.php 1 <?php 2 3 namespace App\Http\Controllers; 4 5 use Illuminate\Http\Request; 6 7 class WelcomeController extends Controller 8 { 9 public function hello() 10 { 11 return 'Hello World'; 12 } 13 } </pre>  <p>Hello World</p> <pre> routes > web.php 1 <?php 2 3 use App\Http\Controllers\WelcomeController; 4 5 Route::get('/hello', [WelcomeController::class, 'hello']); </pre> <p>=> Membuat WelcomeController dengan metode hello(), lalu mengubah route /hello agar tidak lagi menggunakan <i>closure</i></p>
2	Modifikasi routing ke PageController

```

app > Http > Controllers > PageController.php > PageController > about
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6
7 class PageController extends Controller
8 {
9     public function index()
10     {
11         return 'Selamat Datang';
12     }
13
14     public function about()
15     {
16         return 'Nama Saya Faiva - NIM: 244107020036';
17     }
18
19     public function articles($id)
20     {
21         return "Halaman Artikel dengan ID $id";
22     }
23 }

```

```

routes > web.php
1 <?php
2
3 use App\Http\Controllers\PageController;
4
5 Route::get('/', [PageController::class, 'index']);
6 Route::get('/about', [PageController::class, 'about']);
7 Route::get('/articles/{id}', [PageController::class, 'articles']);

```

Selamat Datang

Nama Saya Faiva - NIM: 244107020036

Halaman Artikel dengan ID 5

=> Membuat **PageController** untuk mengelola beberapa *route* dasar sekaligus di dalam satu controller (fungsi index(), about(), dan articles()).

3

Modifikasi menjadi Single Action Controller

```

C:\laragon\www\PWL-2026(main -> origin)
λ php artisan make:controller HomeController --invokable
[INFO] Controller [C:\laragon\www\PWL-2026\app\Http\Controllers\HomeController.php] created successfully.

C:\laragon\www\PWL-2026(main -> origin)
λ php artisan make:controller AboutController --invokable
[INFO] Controller [C:\laragon\www\PWL-2026\app\Http\Controllers\AboutController.php] created successfully.

C:\laragon\www\PWL-2026(main -> origin)
λ php artisan make:controller ArticleController --invokable
[INFO] Controller [C:\laragon\www\PWL-2026\app\Http\Controllers\ArticleController.php] created successfully.

```

```

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

```

```

app > Http > Controllers > AboutController.php > AboutController >
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6
7 class AboutController extends Controller
8 {
9     public function __invoke()
10     {
11         return 'Nama Saya Faiva - NIM: 244107020036';
12     }
13 }

```

```

app > Http > Controllers > ArticleController.php > ArticleController >
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6
7 class ArticleController extends Controller
8 {
9     public function __invoke($id)
10     {
11         return "Halaman Artikel dengan ID $id";
12     }
13 }

```

```

routes > web.php
1 <?php
2
3 use Illuminate\Support\Facades\Route;
4 use App\Http\Controllers\HomeController;
5 use App\Http\Controllers\AboutController;
6 use App\Http\Controllers\ArticleController;
7
8 Route::get('/', HomeController::class);
9 Route::get('/about', AboutController::class);
10 Route::get('/articles/{id}', ArticleController::class);



```

Hasilnya sama seperti Modifikasi langkah 2

=> Memecah logika menjadi Single Action Controller

4

Langkah praktikum 8-11

 PhotoController.php 

```

routes > web.php
1 <?php
2
3 use Illuminate\Support\Facades\Route;
4 use App\Http\Controllers\PhotoController;
5
6 Route::resource('photos', PhotoController::class);

```

```

C:\laragon\www\PWL-2026(main -> origin)
λ php artisan route:list

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      storage/{path} ..... storage.local.upload
PUT           storage/{path} ..... storage.local.upload
GET|HEAD      up .....

```

```

routes > web.php
1 <?php
2
3 use Illuminate\Support\Facades\Route;
4 use App\Http\Controllers\PhotoController;
5
6 Route::resource('photos', PhotoController::class)->only([
7     'index', 'show'
8 ]);
9
10 Route::resource('photos', PhotoController::class)->except([
11     'create', 'store', 'update', 'destroy'
12 ]);

```

Halaman Daftar Photos

Detail Photo ID: 5

	=> Membuat Resource Controller (PhotoController) untuk meng- <i>generate</i> rute CRUD secara otomatis, serta mempraktikkan cara membatasi rute dengan fungsi <code>only()</code> dan <code>except()</code> .
--	--

Praktikum Ke-3 (View)

Langkah	Jawaban/Deskripsi
1	<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></pre>
2	<pre>routes > web.php > Closure 1 <?php 2 3 use Illuminate\Support\Facades\Route; 4 5 Route::get('/greeting', function () { 6 return view('hello', ['name' => 'Faiva']); 7 });</pre>
3	 <p>=> Pembuatan file <i>view</i> menggunakan Blade template (hello.blade.php) dan cara menampilkannya lewat <i>route</i> dengan meneruskan data array ke <i>view</i> tersebut.</p>
4	<p>Langkah 4-7 (View dalam directory)</p>  <pre>routes > web.php > ... 1 <?php 2 3 use Illuminate\Support\Facades\Route; 4 5 Route::get('/greeting', function () { 6 return view('blog.hello', ['name' => 'Faiva']); 7 });</pre>  <p>=> Mempraktikkan cara mengorganisasi file <i>view</i> ke dalam sebuah <i>sub-directory</i> (folder blog) dan mengubah pemanggilan pada <i>route</i> menjadi <code>view('blog.hello')</code>.</p>
5	Langkah 8-10 (Menampilkan View dari Controller)

```

app > Http > Controllers > WelcomeController.php > WelcomeController
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6
7 class WelcomeController extends Controller
8 {
9     public function hello()
10     {
11         return 'Hello World';
12     }
13
14     public function greeting(){
15         return view('blog.hello', ['name' => 'Faiva']);
16     }
17 }

```

```

routes > web.php
1 <?php
2
3 use Illuminate\Support\Facades\Route;
4 use App\Http\Controllers\WelcomeController;
5
6 Route::get('/greeting', [WelcomeController::class, 'greeting']);

```

localhost/PWL-2026/p...
Hello, Faiva

=> Memindahkan logika pemanggilan *view* yang sebelumnya ada di *route* ke dalam sebuah Controller

6

Langkah 11-13 (Meneruskan data ke view)

```

app > Http > Controllers > WelcomeController.php > Welcome
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6
7 class WelcomeController extends Controller
8 {
9     public function hello()
10     {
11         return 'Hello World';
12     }
13
14     public function greeting(){
15         return view('blog.hello')
16             ->with('name','Faiva')
17             ->with('occupation','President');
18     }
19 }

```

```

resources > views > blog > hello.blade.php > ...
1 <!-- View pada resources/views/hello.blade.php -->
2 <html>
3     <body>
4         <h1>Hello, {{ $name }}</h1>
5         <h1>You are {{ $occupation }}</h1>
6     </body>
7 </html>

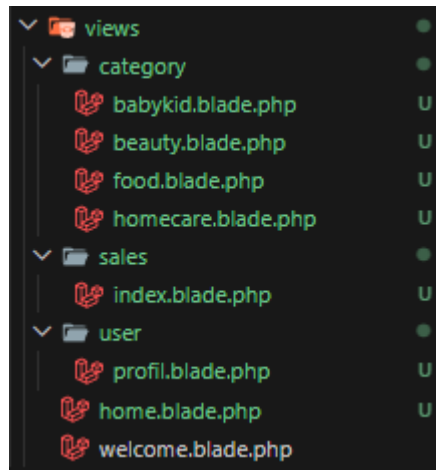
```

localhost/PWL-2026/
Hello, Faiva
You are President

=> Praktik meneruskan data yang lebih spesifik dari Controller ke file *view* menggunakan *method* berantai.

Soal Praktikum

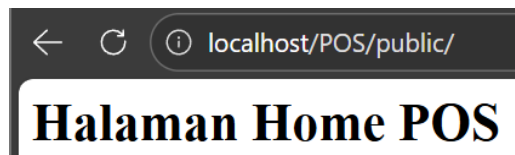
Langkah	Jawaban/Deskripsi
1	<div data-bbox="317 280 1246 472"> <pre> C:\laragon\www λ cd POS C:\laragon\www\POS λ composer create-project laravel/laravel POS Creating a "laravel/laravel" project at "./POS" https://repo.packagist.org could not be fully loaded (exception: loading https://repo.packagist.org/packages.json failed (c 0010 milliseconds), package information was loaded from the cache but may be out of date) Installing laravel/laravel (v12.11.2) </pre> </div> <p>=> Inisialisasi dan pembuatan <i>project</i> Laravel baru dengan nama POS (Point of Sales) menggunakan perintah Composer.</p>
2	<div data-bbox="317 613 1169 1554"> <pre> C:\laragon\www\POS λ php artisan make:controller HomeController INFO: Controller [C:\laragon\www\POS\app\Http\Controllers\HomeController.php] created successfully. C:\laragon\www\POS λ php artisan make:controller ProductController INFO: Controller [C:\laragon\www\POS\app\Http\Controllers\ProductController.php] created successfully. C:\laragon\www\POS λ php artisan make:controller UserController INFO: Controller [C:\laragon\www\POS\app\Http\Controllers\UserController.php] created successfully. C:\laragon\www\POS λ php artisan make:controller SalesController INFO: Controller [C:\laragon\www\POS\app\Http\Controllers\SalesController.php] created successfully. </pre> <div> <div> <pre> app > Http > Controllers > HomeController.php > ... 1 <?php 2 3 namespace App\Http\Controllers; 4 5 use Illuminate\Http\Request; 6 7 class HomeController extends Controller 8 { 9 public function index() 10 { 11 return view('home'); 12 } 13 } </pre> </div> <div> <pre> app > Http > Controllers > ProductController.php > ... 1 <?php 2 3 namespace App\Http\Controllers; 4 5 use Illuminate\Http\Request; 6 7 class ProductController extends Controller 8 { 9 public function foodBeverage() 10 { 11 return view('category.food'); 12 } 13 public function beautyHealth() 14 { 15 return view('category.beauty'); 16 } 17 public function homeCare() 18 { 19 return view('category.homecare'); 20 } 21 public function babyKid() 22 { 23 return view('category.babykid'); 24 } 25 } </pre> </div> </div> <div> <div> <pre> app > Http > Controllers > SalesController.php > ... 1 <?php 2 3 namespace App\Http\Controllers; 4 5 use Illuminate\Http\Request; 6 7 class SalesController extends Controller 8 { 9 public function index() 10 { 11 return view('sales.index'); 12 } 13 } </pre> </div> <div> <pre> app > Http > Controllers > UserController.php > ... 1 <?php 2 3 namespace App\Http\Controllers; 4 5 use Illuminate\Http\Request; 6 7 class UserController extends Controller 8 { 9 public function profile(\$id, \$name) 10 { 11 return view('user.profile', compact('id', 'name')); 12 } 13 } </pre> </div> </div> </div> <p>=> Pembuatan beberapa Controller yang dibutuhkan untuk sistem POS, yaitu HomeController, ProductController, SalesController, dan UserController melalui <i>command-line</i> artisan.</p>
3	<h3>Membuat Route</h3>



=> Menyiapkan *route* dan struktur *view* terpisah untuk berbagai entitas, seperti folder kategori produk, profil *user*, dan halaman transaksi (*sales*).

4

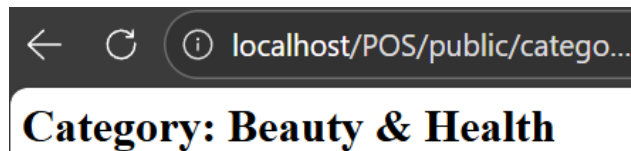
<http://localhost/POS/public/>



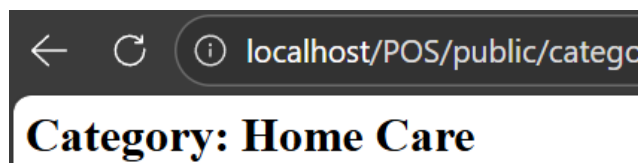
localhost/POS/public/category/food-beverage



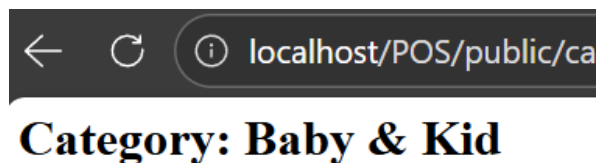
localhost/POS/public/category/beauty-health



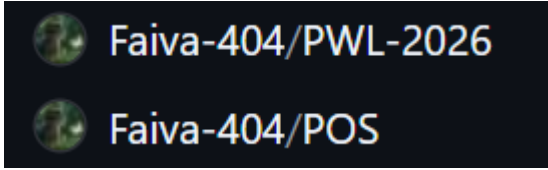


localhost/POS/public/category/home-care



localhost/POS/public/category/baby-kid



	 <p>User Profile</p> <p>ID: 1</p> <p>Name: Faiva</p> <p>localhost/POS/public/sales</p>
5	 <p>Halaman Transaksi POS</p> <p>=> Uji coba <i>running project</i> POS di <i>browser</i> untuk memastikan halaman <i>Home</i>, kategori spesifik (<i>Food & Beverage</i>, <i>Beauty & Health</i>, dll).</p>
	 <p>Upload ke Github</p> <p>Faiva-404/PWL-2026</p> <p>Faiva-404/POS</p>