

INTRODUCTION

LARAVEL 8

Muhammad Nugraha, M.Eng



CONTENTS

Lore ipsum dolor sit amet,



01. Introduction

Pengenalan laravel 8

02. Instalasi & Konfigurasi

Persiapan sistem sebelum instalasi, download semua yang di butuhkan oleh sistem, dan konfigurasi.

03. Struktur Folder

Struktur folder dari Laravel 8

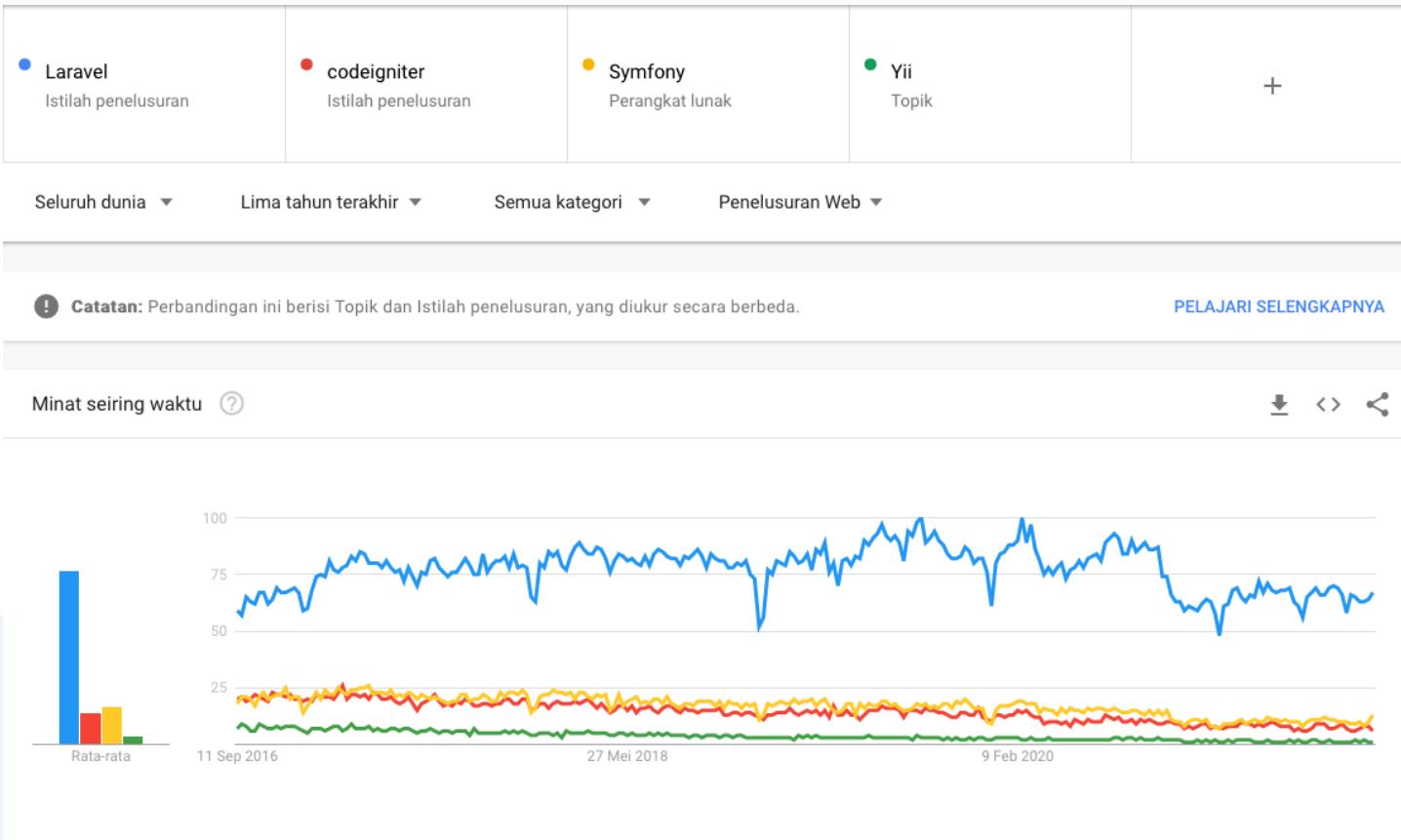
04. Route & View

cara menggunakan route dan view di laravel 8

Pendahuluan

O Introduction

- ❑ Laravel merupakan proyek open source yang dirintis oleh Taylor Otwell bertujuan untuk mengembangkan aplikasi berbasis web dengan arsitektur MVC (Model-View-Controller).
- ❑ Laravel adalah framework php paling popular dan banyak digunakan di dunia.

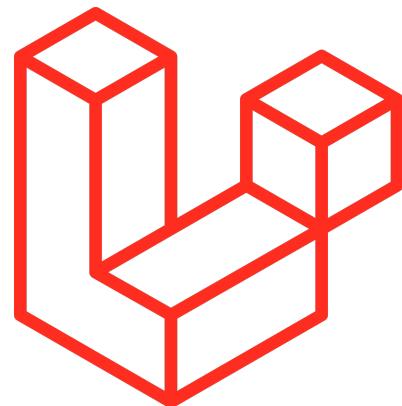




Mengapa Laravel?

- Sintaks yang ringkas dan elegan
- Open Source
- Ekosistem Bagus
- Keamanan Bagus
- Modern

Instalasi & Konfigurasi



Persiapan System

- Git
- Composer
- Web Server (Apache, Nginx, Valet)
- MySql (PhpMyadmin / Adminer)
- VSCode
- Browser



Laravel 8.0 - Latest ▾

Server Requirements.

PHP >= 7.3.0 ✓ (7.3.7)
OpenSSL PHP Extension ✓
PDO PHP Extension ✓
Mbstring PHP Extension ✓
Tokenizer PHP Extension ✓
XML PHP Extension ✓
CTYPE PHP Extension ✓
JSON PHP Extension ✓
BCmath PHP Extension ✓
magic_quotes_gpc: ✓ (value:)
register_globals: ✓ (value:)
session.auto_start: ✓ (value: 0)
mbstring.func_overload: ✓ (value: 0)



Install Composer (linux/macOS)

- Jalankan perintah berikut di terminal:

```
php -r "copy('https://getcomposer.org/installer', 'composer-setup.php');"  
php -r "if (hash_file('sha384', 'composer-setup.php') === '756890a4488ce90  
24fc62c56153228907f1545c228516cbf63f885e036d37e9a59d27d63f46af1d4d07ee0f76  
181c7d3') { echo 'Installer verified'; } else { echo 'Installer corrupt';  
unlink('composer-setup.php'); } echo PHP_EOL;"  
php composer-setup.php  
php -r "unlink('composer-setup.php');"
```

- Setelah selesai download composer-setup.php maka pindahkan file tersebut dengan perintah:

```
sudo mv composer.phar /usr/local/bin/composer
```

nb: jika menggunakan **windows** download composer dan install seperti biasa

Install Web Server (**Apache**)

- ❑ Untuk menggunakan Apache bisa dengan menginstall **XAMPP** atau paketan web server lainnya yang didalamnya sudah termasuk MySql, php, dan phpMyAdmin.
- ❑ Untuk pengguna linux/mac bisa juga install secara terpisah dengan cara menginstall Apachenya saja.

nb: Sebenarnya tidak menginstall web server juga tidak masalah, karena didalam Laravel 8 sendiri sudah ada **artisan** sebagai web servernya



Install Web Server (**Valet**)

- Valet ini digunakan khusus dengan system operasi macOS

```
composer global require laravel/valet
```

```
valet install
```

```
valet use php@7.3
```

- Setelah selesai di install maka masuk direktori tempat file web disimpan dan ketikan:

```
valet park
```

nb: setiap direktori yang ada di park tersebut akan menjadi nama domain dengan akhiran **.test**

Misal pada direktori tersebut ada nama direktori **Latihan**, maka untuk mengaksesnya ketik di browser **Latihan.test**

O Install Laravel 8

□ Install Laravel 8 dengan menggunakan composer:

```
composer create-project laravel/laravel=^8.0 latihan --prefer-dist
```

Ket:

composer create-project: Perintah bikin projek baru
laravel/laravel=^8.0: Nama paket yang di download
latihan: Nama folder atau direktori hasil downloadan

□ Membuka Laravel 8

`http://localhost/latihan/public` → apache

`latihan.test` → valet

`php artisan serve` → artisan

kemudian buka di browser dengan alamat:

`http://127.0.0.1:8000/`



Documentation

Laravel has wonderful, thorough documentation covering every aspect of the framework. Whether you are new to the framework or have previous experience with Laravel, we recommend reading all of the documentation from beginning to end.



Laracasts

Laracasts offers thousands of video tutorials on Laravel, PHP, and JavaScript development. Check them out, see for yourself, and massively level up your development skills in the process.



Laravel News

Laravel News is a community driven portal and newsletter aggregating all of the latest and most important news in the Laravel ecosystem, including new package releases and tutorials.



Vibrant Ecosystem

Laravel's robust library of first-party tools and libraries, such as [Forge](#), [Vapor](#), [Nova](#), and [Envoyer](#) help you take your projects to the next level. Pair them with powerful open source libraries like [Cashier](#), [Dusk](#), [Echo](#), [Horizon](#), [Sanctum](#), [Telescope](#), and more.



[Shop](#)



[Sponsor](#)

Laravel v8.58.0 (PHP v7.3.30)

Struktur Folder, Route, & View



Struktur Folder Laravel 8

```
app
├── Console
├── Exceptions
└── Http
    ├── Controllers
    └── Models
    ├── Middleware
    └── Kernel.php
├── Models
├── Providers
├── bootstrap
├── config
├── database
├── public
├── resources
│   ├── css
│   ├── js
│   ├── lang
│   └── views
└── routes
└── storage
└── tests
└── vendor
```

```
.editorconfig
.env
.env.example
.ATTRIBUTES
.gitignore
.styleci.yml
README.md
artisan
composer.json
composer.lock
package.json
phpunit.xml
server.php
webpack.mix.js
```

Konsep MVC

MVC merupakan kependekan dari Model View Controller:

Model *app/Models*

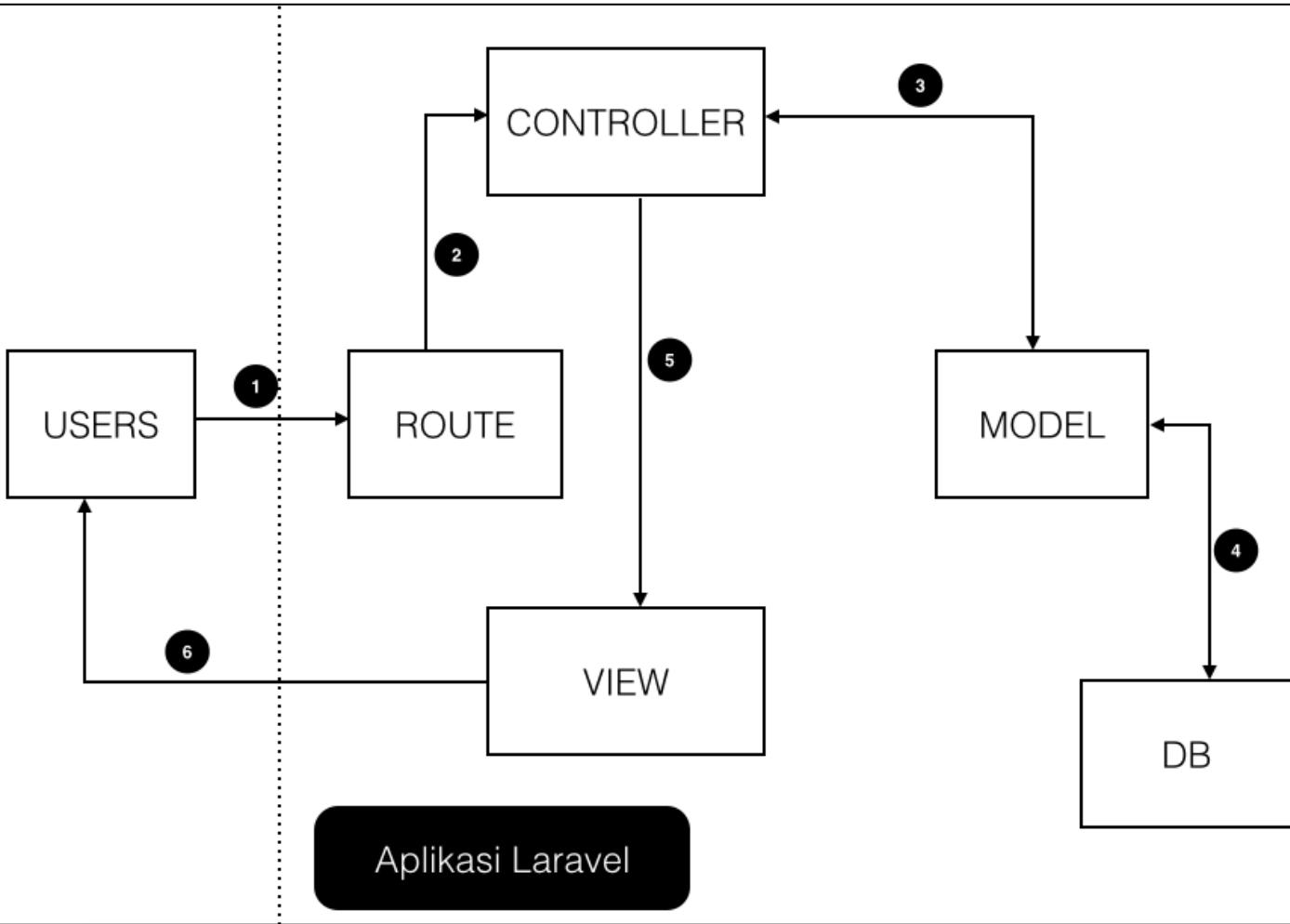
Merupakan komponen dalam aplikasi yang bertugas untuk mengelola data.

View *resources/views*

Merupakan komponen dalam aplikasi kita yang bertanggungjawab untuk membuat tampilan / interface untuk pengguna. Sumber data didapat dari model yang didapatkan melalui controller. Tidak berinteraksi langsung dengan database. View juga menangkap interaksi dari pengguna yang akan diteruskan ke aplikasi.

Controller *app/Http/Controllers*

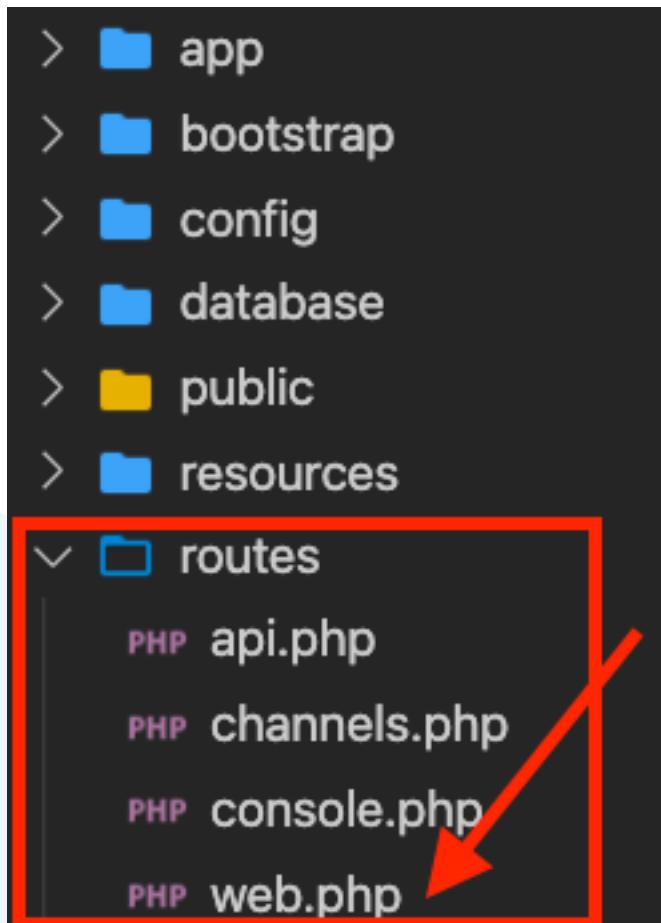
Merupakan komponen dalam aplikasi kita yang bertanggungjawab untuk menerima input dan memberikan output, atau dalam dunia web kita lebih mengenal dengan istilah request dan response. Controller bertugas untuk menerima request, kemudian memprosesnya dengan memberikan response baik berupa data atau view berisi data dari model.



1. User mengakses aplikasi melalui route tertentu.
2. Route tersebut oleh aplikasi telah dipetakan ke controller action.
3. Controller action akan menggunakan model untuk mengakses data. Atau langsung mengembalikan view tanpa data (langsung ke step 5).
4. Model berinteraksi ke database untuk mendapatkan data atau menyimpan data.
5. Setelah berhasil mendapatkan data melalui model, controller akan mengembalikan sebuah view sekaligus data jika ada.
6. View tersebut pada akhirnya yang dilihat oleh user.

O Routing

- ❑ Routing adalah proses menentukan arah atau rute yang harus jalankan.
- ❑ Intinya tugas utama routing untuk mengatur proses yang dilakukan laravel ketika ada request dari pengguna aplikasi.



```
1 Route::get('/', function () {  
2     return view('welcome');  
3 });
```

O Routing

□ Sintaks:

```
Route::http_method("path", "action");
```

http_method : get, post, put, delete

path : alamat pengaksesan di address browser

action : action ketika di akses method dan pathnya

O View

- View bertanggungjawab untuk memberikan tampilan ke user.
- Jika kita ingin meletakkan kode html, css dan javascript di view lah tempatnya.

The screenshot shows a code editor interface with a sidebar titled 'OPEN EDITORS' and a main panel displaying a Blade template file.

OPEN EDITORS

- LARAVEL8
 - app
 - bootstrap
 - config
 - database
 - public
 - resources
 - css
 - js
 - lang
 - views
 - coba.blade.php
 - welcome.blade.php

resources > views > coba.blade.php > html

```
1  <!DOCTYPE html>
2  <html lang="en">
3
4  <head>
5      <meta charset="UTF-8">
6      <meta http-equiv="X-UA-Compatible" content="IE=edge">
7      <meta name="viewport" content="width=device-width, initial-scale=1.0">
8      <title>Halaman View</title>
9  </head>
10
11 <body>
12     <h1>Halaman View yang akan di panggil dari Routes</h1>
13 </body>
14
15 </html>
```

A red box highlights the 'views' folder in the sidebar, and a red arrow points from the bottom of this box to the 'coba.blade.php' file in the main panel.

- Untuk mengakses view dari router caranya adalah seperti berikut:

```
● ● ●  
1 Route::get('/', function () {  
2     return view('welcome');  
3 });  
4  
5 Route::get('/tes', function () {  
6     return view('coba');  
7 });
```

path untuk pemanggilan di browser

nama file di view yang dipanggil

`http://localhost/latihan/public/tes`

Halaman View yang akan dipanggil dari Routes



Route berdasarkan jenis HTTP Method

GET

GET method merupakan method paling dasar dan umum digunakan. Jika kita mengakses alamat URL sebuah di browser, maka itu termasuk contoh penggunaan GET method.

POST

POST method umumnya digunakan untuk melakukan proses submit data ke aplikasi. Sehingga sering kita temukan penggunaannya pada form.

PUT

PUT method mirip dengan POST method yaitu digunakan juga untuk mengirimkan data ke aplikasi baik dari form ataupun ajax request. Bedanya PUT method lebih sering digunakan untuk mengirimkan data yang bersifat UPDATE bukan CREATE. Misalnya di form edit.

DELETE

DELETE method sesuai namanya digunakan untuk melakukan penghapusan sebuah data di aplikasi tertentu. Biasanya kita akan mengirimkan ID data tertentu yang ingin kita hapus baik melalui form atau ajax request.

O Route berdasarkan jenis HTTP Method (Contoh)

□ GET

```
Route::get("/products/list", [ProductController::class, 'list']);
```

□ POST

```
Route::post("/products/create", [ProductController::class, 'create']);
```

□ PUT

```
Route::put("/products/{id}", [ProductController::class, 'update']);
```

□ DELETE

```
Route::delete("/products/{id}", [ProductController::class, 'delete']);
```

O Latihan

- Buatlah tutorial untuk menginstall dan mengconfigurasi Laravel 8. Mulai dari install aplikasi-aplikasi pendukungnya, configurasinya, sampai Laravel 8 siap untuk digunakan.
- Setelah laravel 8 siap digunakan, buatlah 5 buah file view(bebas) dan buat route untuk memanggil ke 5 view tersebut.

Ket: Idealnya **view** di panggil dari **controller** bukan dari **route**, tapi hal ini bisa kita lakukan jika kita ingin menampilkan **view** tanpa data yang kompleks atau **view** bersifat statis

Thanks !