

Pada dasarnya Laravel memiliki alur utama / workflow seperti berikut

Route > controller + eloquent (model) > view

Route ---> routes/web.php

Berfungsi untuk mengatur URL yang mana dan file apa yang dieksekusi ketika URL itu dipanggil.

Untuk melihat list route, dapat menggunakan command `php artisan route:list` di dalam folder Laravel. Pastikan buka di terminal terpisah dengan VS Code agar bisa dilihat Fullscreen.

MINGW64: c:/laragon/www/clinic

```
Arix Wap@WAP-PC MINGW64 /c:/laragon/www/clinic (master)
$ php artisan route:list
```

Domain	Method	URI	Name	Action	Middleware
	GET HEAD	/		Closure	web
	GET HEAD	api/user		Closure	api,auth:api
	GET HEAD	home	home	App\Http\Controllers\HomeController@index	web,auth
	GET HEAD	login	login	App\Http\Controllers\Auth\LoginController@showLoginForm	web,guest
	POST	login		App\Http\Controllers\Auth\LoginController@login	web,guest
	POST	logout	logout	App\Http\Controllers\Auth\LoginController@logout	web
	POST	password/confirm		App\Http\Controllers\Auth\ConfirmPasswordController@confirm	web,auth
	GET HEAD	password/confirm	password.confirm	App\Http\Controllers\Auth\ConfirmPasswordController@showConfirmForm	web,auth
	POST	password/email	password.email	App\Http\Controllers\Auth\ForgotPasswordController@sendResetLinkEmail	web
	POST	password/reset	password.update	App\Http\Controllers\Auth\ResetPasswordController@reset	web
	GET HEAD	password/reset	password.request	App\Http\Controllers\Auth\ForgotPasswordController@showLinkRequestForm	web
	GET HEAD	password/reset/{token}	password.reset	App\Http\Controllers\Auth\ResetPasswordController@showResetForm	web
	GET HEAD	patient	patient.index	App\Http\Controllers\PatientController@index	web,auth
	POST	patient	patient.store	App\Http\Controllers\PatientController@store	web,auth
	GET HEAD	patient/create	patient.create	App\Http\Controllers\PatientController@create	web,auth
	GET HEAD	patient/{patient}	patient.show	App\Http\Controllers\PatientController@show	web,auth
	PUT PATCH	patient/{patient}	patient.update	App\Http\Controllers\PatientController@update	web,auth
	DELETE	patient/{patient}	patient.destroy	App\Http\Controllers\PatientController@destroy	web,auth
	GET HEAD	patient/{patient}/edit	patient.edit	App\Http\Controllers\PatientController@edit	web,auth
	POST	register		App\Http\Controllers\Auth\RegisterController@register	web,guest
	GET HEAD	register	register	App\Http\Controllers\Auth\RegisterController@showRegistrationForm	web,guest

Ini dia penjelasan table route nya

1. Method

Jenis method yang dipanggil melalui URL. Disana ada 4 jenis method, yaitu:

GET|HEAD, POST, PUT|PATCH, DELETE

Selama ini kita mengenal hanya ada method **GET** dan **POST**. Ye khan?

Tapi ternyata ada lagi 2 method yaitu **PUT/PATCH** dan **DELETE**.

Bedanya **PUT/PATCH** digunakan untuk mengedit / mengubah data di dalam Laravel.

Tidak seperti **POST** yang hanya menyimpan data. Method **DELETE** sendiri untuk menghapus data.

2. URI

inilah URL yang tampil di browser. Jika nama URI disana adalah `patient/create`, maka di browser akan tampil menjadi `localhost/patient/create`.

Di URI Patient ada nama `patient/{patient}`, nama di dalam kurawal itu artinya **variable**. Jadi bisa berupa angka ID atau lainnya

Jika kita perhatikan, ada beberapa URI yang memiliki URL yang sama, misalnya URI `patient/{patient}`. Namun apa yang membedakannya?? Method lah yang membedakannya. Jadi sebuah URI dipanggil dengan sebuah method yang dipilih, akan diarahkan ke file action yang dipilih sesuai urutan garisnya

3. Name

Setelah melihat URI diatas, pasti kamu berpikir bagaimana cara memanggilnya berdasarkan method yang diinginkan khan? Laravel sudah menyediakan hal ini. kamu hanya perlu menggunakan fungsi `route('nama.rutenya')`.

Misalnya kita lihat `patient.store`, jika kita mengetik `route('patient.store')`, maka akan dibuat URL : `localhost/patient` dengan method post

NAMUN, perlu diingat. Jika memilih route yang memiliki URI `{nama_variable}`, kamu harus memasukkan variable di parameter kedua.

Contoh : `patient.update`

Pemanggilan route nya menjadi `route('patient.update', $idPatient)`

Contoh lengkapnya bisa dilihat di file saja nanti

4. Action

Ini adalah directory file controller yang dipanggil oleh route tersebut. symbol `@` adalah nama function di dalam file controller tersebut

5. Middleware

Middleware adalah sejenis coding yang dijalankan Laravel sebelum atau sesudah controller di eksekusi.

Disana ada beberapa jenis middleware:

<code>web</code>	: url bisa diakses public
<code>api, auth:api</code>	: url digunakan utk API (Skip dulu krn tidak akan digunakan disini)
<code>web, guest</code>	: url bisa diakses public
<code>web, auth</code>	: url hanya bisa diakses jika login

Mungkin untuk sekarang kita hanya akan cukup menggunakan middleware **Auth**

lewat `route/web.php`. Ada contohnya disana.

Untuk cara kerja dan pengembangannya nanti saja dipelajari jika sudah memahami dasar dasar laravel

Controller ---> app/Http/Controllers/

Inilah file utama dimana kamu akan memilih data, mengolah data, memilih view untuk menampilkan data di dalam web.

Tadi sudah lihat di route khan? Route mana akan memanggil controller dengan function yang mana. Disinilah sekarang gunanya function tersebut.

Laravel default memiliki 7 function / method :

1. **public function index()**
biasanya untuk pengolahan dan penampilan seluruh data
2. **public function create()**
biasanya untuk menampilkan form input data
3. **public function store()**
untuk proses penyimpanan data dari form di function create
4. **public function show()**
biasanya untuk menampilkan halaman detail dari sebuah data
5. **public function edit()**
biasanya untuk menampilkan form edit dari sebuah data yang dipilih
6. **public function update()**
untuk proses update data dari hasil input form edit
7. **public function destroy()**
untuk menghapus data

Sebenarnya kita sendiri bisa membuat custom function dengan proses custom yang kita inginkan. Namun Laravel sudah menyediakan function dasar yang umum sering digunakan. Untuk sekarang kita ikuti dulu standar yang diberikan Laravel.

Di dalam function controller inilah kita akan menggunakan **ELOQUENT** / Model nya si Laravel. ELOQUENT lah yang bertugas menghubungkan database dengan controller.

Basically, satu Eloquent hanya mewakili satu table saja. Tapi untuk keperluan relationship tetap dapat dilakukan, nanti di sesi 2 saja. Tapi jika kepo bisa dibaca di dokumentasi Laravel.

Untuk pahami dulu cara menggunakannya. Oh ya, contohnya ada di **PatientController** dimana dia berhubungan dengan Eloquent Patient.

Sama seperti Migration, Eloquent dan Controller juga bisa dibuat dengan menggunakan command php artisan

Setelah mengolah data dengan eloquent, barulah data tersebut ditampilkan di view

Views ---> resources/views/

HTML CSS, JS semua dipanggil disini.

Data yang kamu olah di controller akan ditampilkan disini juga.

Yang perlu dipahami adalah, LARAVEL menggunakan BLADE sebagai template viewnya.

Example : nama_view.blade.php

Di dalam folder view ada folder **auth, layouts, patient**

folder layout layaknya seperti puzzle. Didalamnya ada blade template utama yang isi contentnya bisa kita ubah.

Jika dilihat di **PatientController**, kita hanya melihat return **view('patient.create')**.

Itu artinya kita memanggil file blade di dalam folder **patient/create.blade.php**

Namun bagaimana kita bisa memilih layout yang diinginkan?

Itu diatur di dalam file create.blade.php itu sendiri. Di atas line pertama ada fungsi blade **@extend('layouts.admin')**

Artinya kita menggunakan layout admin di dalam view tersebut.

Coba kita buka file layout.admin.php

di dalamnya ada fungsi blade bernama **@yield('content')**. Di dalam sana lah content dari view kita ditampilkan.

Jadi 1 layout bisa digunakan berkali kali

STYLING CSS ---> resources/sass/

Kepikiran ga buat gimana caranya bikin file CSS di Laravel ?

Disini kita menggunakan SASS / SCSS, sedikit lebih kece dari CSS

Namun kamu tidak perlu terlalu dalam mempelajari SASS, jika belum mengerti kamu bisa menggunakan coding CSS di dalam SASS.

Jika ingin tetap mencoba belajar SASS, bisa disini : <https://sass-lang.com/guide>

Di dalam folder itu ada 4 files :

- `app.scss` : ini main file SASS nya
- `functions.scss` : tempat meletakkan mixin / functions SASS
- `variables.scss` : tempat variable SASS,
- `theme.scss` : style utama mu, disini kamu akan sering bekerja

PERLU DIKETAHUI. SASS memerlukan NPM untuk mengcompile file scss menjadi css.

(Browser tidak akan bisa membaca file scss langsung)

Karena itu sebelum mengedit `theme.scss`, pastikan menjalankan command `npm run watch`

CATATAN : Ketika pertama kali inisial Laravel, pastikan melakukan `npm install`. Jika tidak `npm run watch` kemungkinan akan error karena file `node_modules` belum lengkap

Untuk sekarang itu saja dulu..