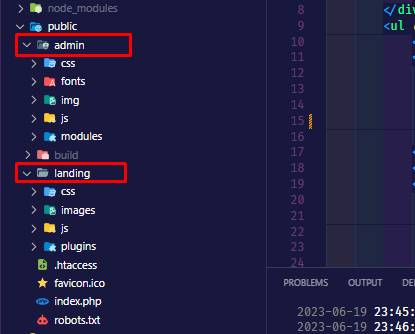
**Simple Templating Laravel**

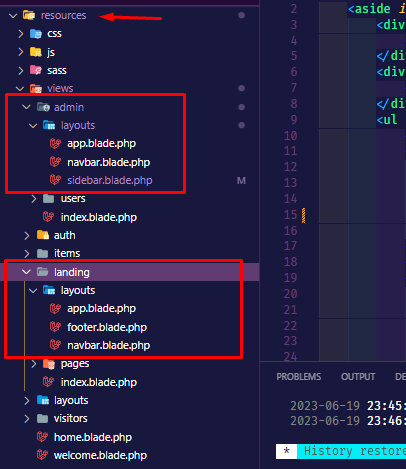
1. Tempatkan theme/template pada public/nama-theme, nama theme sesuai yang diinginkan dan berisi struktur dari theme tersebut, dalam hal ini saya membuatkan landing untuk landing page, dan admin untuk dashboard admin.



Gambar struktur folder theme

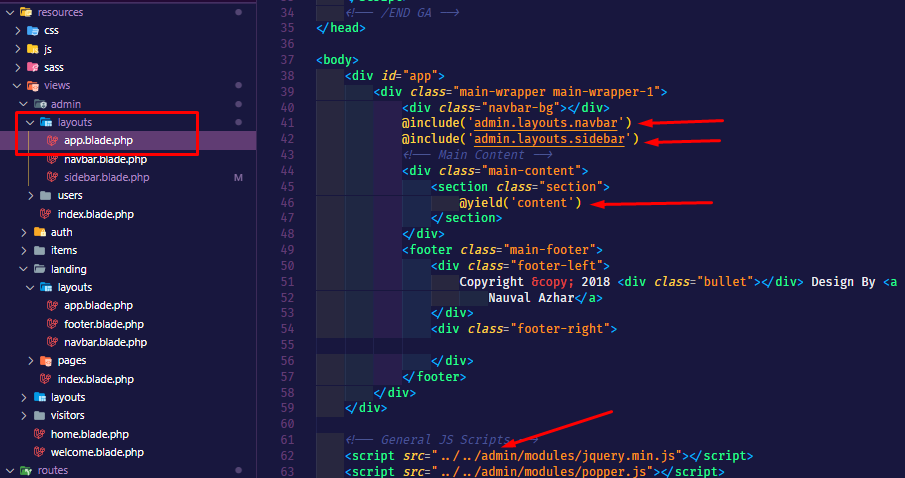
1. Pada folder resources/views buat masing-masing untuk templatingnya, dalam hal ini bebas menamakan tapi untuk lebih baiknya samakan saja dengan folder theme, saya menggunakan landing dan admin pada komponen viewsnya

* Dalam folder landing dan admin masing-masing buatlah folder layouts untuk menentukan layout dari masing-masing template
* Pada folder layouts, buatlah file blade yang memisahkan antara header, footer, navbar, maupun sidebar tergantung struktur dari theme tersebut.
* Pada landing saya membuat layouts app.blade.php untuk menentukan templating yang dipakai, terserah mau menamakan header atau app, saya lebih memilih app karena didalam app semua template akan menjadi satu, lalu memisahkan navbar dan footer dengan templating navbar.blade.php dan footer.blade.php.
* Pada folder admin juga membuat hal sama tapi dengan penamaan berbeda sesuai struktur theme tersebut, saya memakai app, navbar dan sidebar.

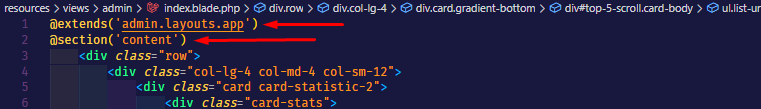


Gambar struktur folder layouts untuk pemakaian theme

1. Untuk penggunaan blade template, perhatikan pada gambar dibawah.



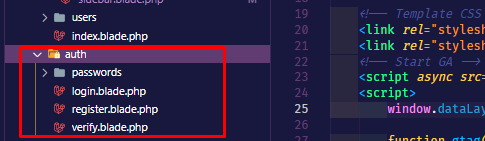
* Gunakan metode include pada php untuk mengambil layout navbar begitu juga dengan sidebar sesuai struktur susunan folder bladenya, dalam hal ini admin/layouts.
* Gunakan yield untuk menunjukan bahwa disini adalah letak untuk setiap konten, jadi untuk konten seterusnya hanya menggunakan extend app lalu section content pada setiap konten yang dibuat.



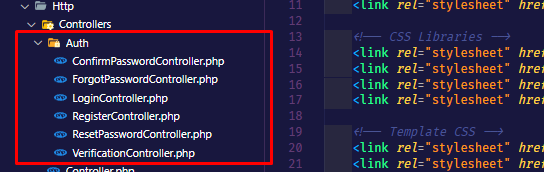
* Lalu untuk mengambil styiling maupun komponen JS pendukung theme tersebut, cukup menggunakan ../../nama-theme atau juga bisa {{ asset(‘/nama-theme’) }} atau juga bisa {{ url(‘/’) }}/nama-theme, dalam hal ini saya menggunakan ../../nama-theme

**Membuat Auth Scaffold Laravel**

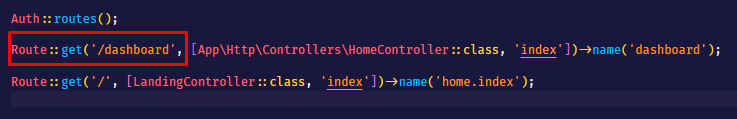
1. Untuk membuat auth sebenarnya ada berbagai macam cara, yaitu dengan pembuatan sendiri dari 0 (from scratch) atau membuat dengan package, package yang bisa digunakan adalah seperti laravel Jetstream, laravel/ui dsb. Dalam hal ini saya menggunakan laravel/ui karena simple dan mudah dicustom.
2. Untuk menginstall package laravel/ui gunakan composer dengan mengetikkan ***composer require laravel/ui****.*
3. Untuk membuat scaffold otomatis yang disediakan oleh laravel/ui ketikkan ***php artisan ui bootstrap –auth***maka akan membuat secara otomatis scaffold dan kita tidak perlu membuat dari awal, berikut adalah hasil generate scaffold laravel/ui



* Laravel/ui membuat otomatis halaman login, register dll. Yang nantinya akan mudah kita mengcustom sendiri halaman tersebut.



* Laravel/ui juga akan meng-generate otomatis controller yang dibutuhkan, tapi kita perlu mengkustom apabila jika berhasil login akan kemana kita diarahkan, dalam hal ini kita mengarahkan ke dashboard admin, maka ubah **Providers/RouteServiceProvider.php** ganti **public const HOME = '/home';** menjadi **public const HOME = '/dashboard';** dan pastikan route dashboard sudah dibuat

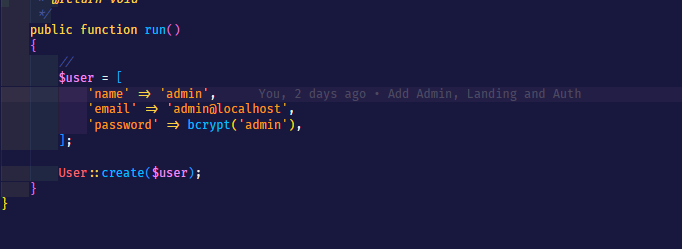


1. Buat batasan pada route, seperti gambar dibawah



Gunakan middleware auth untuk membatasi user yang mana jika login atau sudah auth maka bisa ke dashboard, pengguna.

1. Membuat seeder untuk pembuatan akun admin, untuk membuat seeder gunakan command pada terminal ***php artisan make:seeder AdminSeeder,*** maka laravel akan membuatkan file seeder otomatis pada folder database/seeders. Pembuatan akun admin lebih mudah dibuat secara seeder karena semisal kita ada role dan permission lebih mudah untuk seed data admin.



Pembuatan akun dalam seeder seperti gambar diatas, yang nantnya kita login melalui akun tersebut.

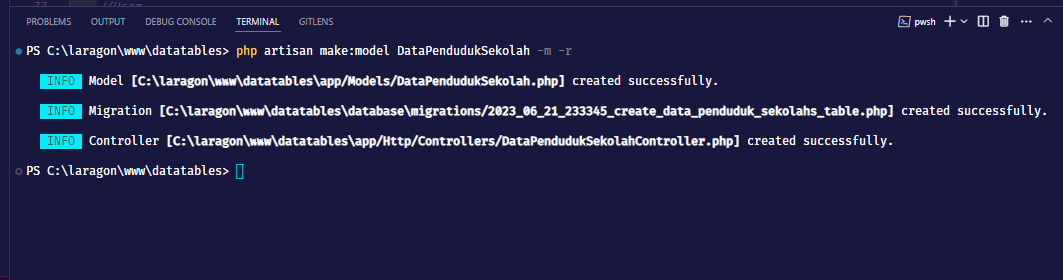
Lalu untuk memanggil seeder, edit pada DatabaseSeeder.php lalu pada function run isikan ***$this->call(NamaSeeder::class);*** dalam hal ini namanya adalah AdminSeeder seperti gambar dibawah



Setelah kita membuat auth scaffolding dan seeder, maka kita harus menjalankan migrate dengan seed tersebut, gunakan command ***php artisan migrate –seed*** atau jika ingin drop semua table lalu mengganti yang baru gunakan ***php artisan migrate:fresh –seed***.

**Membuat Model, Mogration dan Controller**

1. Sebelum masuk pada CRUD, kita wajib mendesain databasenya terlebih dahulu, untuk membuat migrasi database gunakan perintah ***php artisan make:migration NamaMigrasi*** maka akan dibuatkan otomatis dengan nama migrasi, namun jika ingin membuat migrasi lengkap dengan Model dan Controller maka gunakan perintah ***php artisan make:model NamaMigrasi –m –r***. perintah –m langsung membuat Model dengan nama migrasi, lalu –r adalah membuat Controller dengan resource dan model binding. Contohnya seperti gambar dibawah.



Sesuaikan skema database seuai yng diinginkan, semisal saya akan membuat table kecamatan yang didalamnya terdapat kolom id (autoincrement), nama\_kecamatan dan timestamps (created\_at, updated\_at).

