

**MODUL MEMBUAT NOTIFIKASI MENGGUNAKAN LARAVEL
FRAMEWORK PEMROGRAMAN WEB**



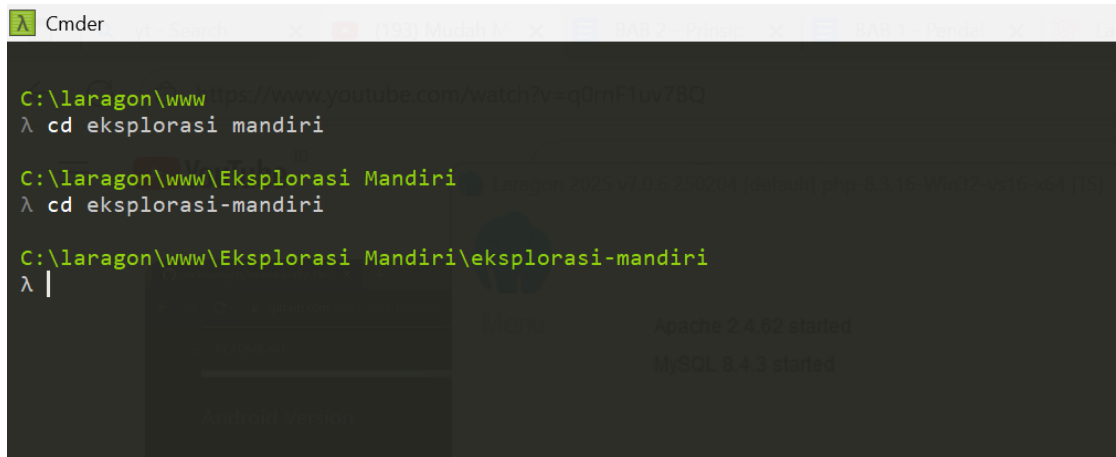
NAMA : Aprillia Oktan
NPM : 2310631170004
KELAS : 5D

**PROGRAM STUDI INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS SINGAPERBANGSA KARAWANG**

2025

Pada percobaan ini dilakukan eksplorasi mandiri untuk memahami proses pembuatan dan pengaturan notifikasi pada Laravel. Setiap langkah dikerjakan secara berurutan dengan tujuan agar mekanisme konfigurasi hingga pengiriman notifikasi dapat dipahami dan diterapkan dengan baik. Penjelasan berikut disusun secara ringkas namun tetap sistematis.

1.



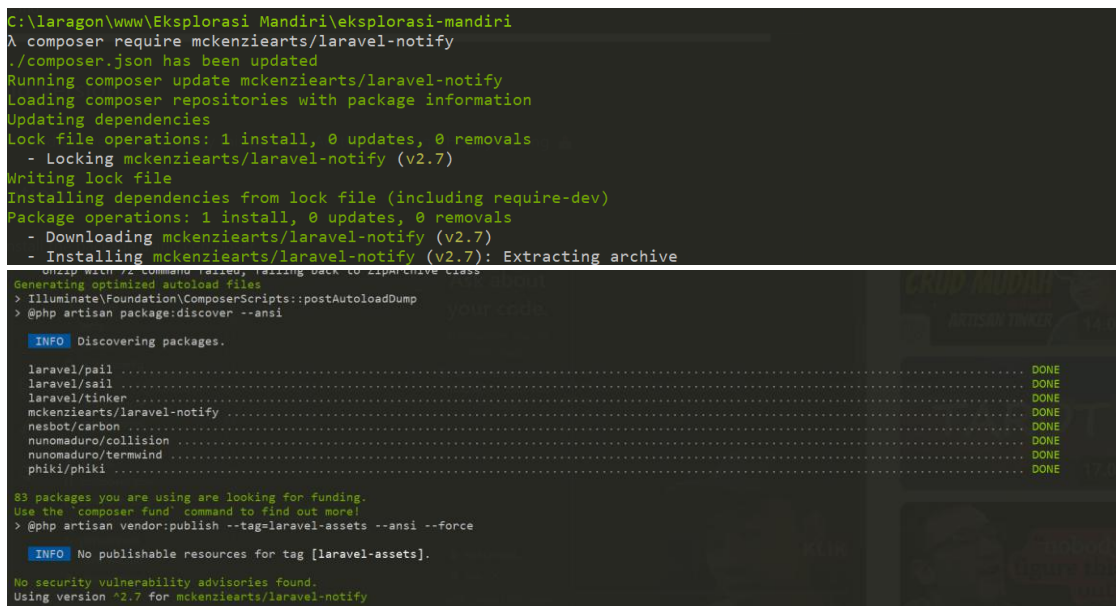
```
C:\laragon\www
λ cd eksplorasi mandiri

C:\laragon\www\Eksplorasi Mandiri
λ cd eksplorasi-mandiri

C:\laragon\www\Eksplorasi Mandiri\eksplorasi-mandiri
λ |
```

Langkah awal, arahkan ke direktori proyek Laravel yang baru saja dibuat. Proses ini dapat dilakukan melalui terminal di VS Code yang diarahkan ke folder proyek tersebut, atau menggunakan terminal bawaan Laragon. Pada contoh ini, saya menggunakan terminal Laragon.

2.



```
C:\laragon\www\Eksplorasi Mandiri\eksplorasi-mandiri
λ composer require mckenziearts/laravel-notify
./composer.json has been updated
Running composer update mckenziearts/laravel-notify
Loading composer repositories with package information
Updating dependencies
Lock file operations: 1 install, 0 updates, 0 removals
- Locking mckenziearts/laravel-notify (v2.7)
Writing lock file
Installing dependencies from lock file (including require-dev)
Package operations: 1 install, 0 updates, 0 removals
- Downloading mckenziearts/laravel-notify (v2.7)
- Installing mckenziearts/laravel-notify (v2.7): Extracting archive
Generating optimized autoload files
> Illuminate\Foundation\ComposerScripts::postAutoloadDump
> @php artisan package:discover --ansi

INFO: Discovering packages.

laravel/pail ..... DONE
laravel/sail ..... DONE
laravel/tinker ..... DONE
mckenziearts/laravel-notify ..... DONE
nesbot/carbon ..... DONE
nunomaduro/collision ..... DONE
nunomaduro/termwind ..... DONE
phiki/phiki ..... DONE

83 packages you are using are looking for funding.
Use the 'composer fund' command to find out more!
> @php artisan vendor:publish --tag=laravel-assets --ansi --force

INFO: No publishable resources for tag [laravel-assets].

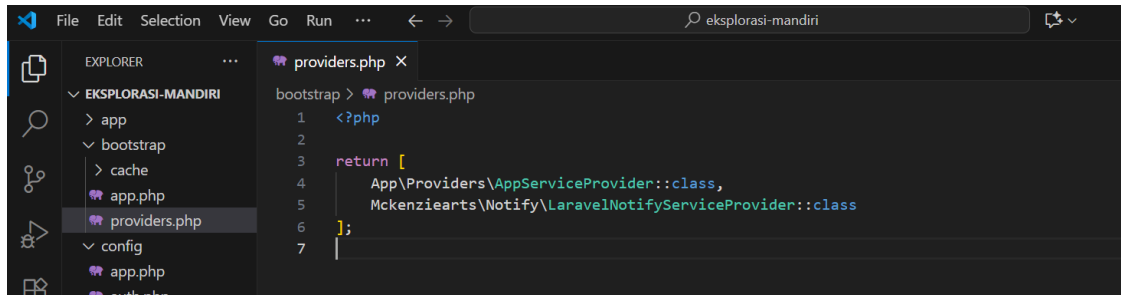
No security vulnerability advisories found.
Using version ^2.7 for mckenziearts/laravel-notify
```

Jalankan perintah berikut di direktori proyek:

‘composer require mckenziearts/laravel-notify’

Perintah ini menginstal paket Laravel Notify (oleh mckenziearts) yang menyediakan flash/toast notifications yang cantik dan mudah dipakai pada tampilan (frontend) — jadi ini berguna untuk menampilkan notifikasi singkat di UI, bukan menggantikan sistem notifikasi server-side bawaan Laravel.

3.



Selanjutnya ke VS Code lalu ke file 'providers.php' di folder bootstrap yang dan tambahkan

'Mckenziearts\Notify\LaravelNotifyServiceProvider::class'

Tambahan ini memastikan Laravel dapat memuat service provider Laravel Notify secara otomatis saat aplikasi dijalankan, sehingga fitur notifikasi bisa digunakan di seluruh bagian aplikasi.

4.



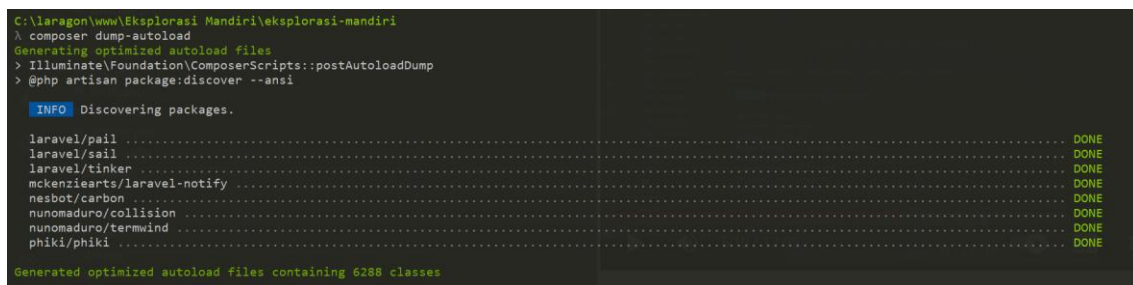
Kembali lagi ke terminal laragon dan jalankan

'php artisan vendor:publish --

provider="Mckenziearts\Notify\LaravelNotifyServiceProvider"'

Proses ini mempublikasikan aset-aset bawaan paket Laravel Notify ke direktori aplikasi, seperti file konfigurasi notify.php dan folder vendor pada public, sehingga kita dapat menyesuaikan aset notifikasi sesuai kebutuhan.

5.



Lalu jalankan perintah

'composer dump-autoload'

Proses ini berfungsi untuk mengoptimalkan autoload file PHP dan menemukan package yang baru diinstal, termasuk Laravel Notify. Hasilnya terlihat semua paket berhasil terdeteksi dengan status DONE yang berarti autoload sudah berhasil diperbarui.

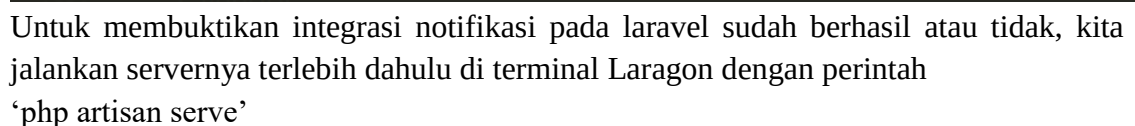
6.

Kembali ke VS Code, lalu buat file blade baru namanya bebas sebagai contoh 'notif_eksplorasi.blade.php' yang berfungsi sebagai view atau tampilan notifikasi. Pada file ini sudah dimasukkan '@notifyCss' dan '@notifyJs' agar komponen Laravel Notify bisa berfungsi. Selain itu, terdapat kode '<x-notify::notify />' yang menjadi penampung (container) notifikasi di halaman tersebut, dengan latar belakang biru sesuai style CSS yang diberikan.

7.

Langkah selanjutnya membuat route baru namanya bebas, namun disini saya menggunakan nama yang sama dengan file blade tadi yaitu 'notif_eksplorasi' yang memanggil fungsi 'notify() ->success ()' untuk memunculkan notifikasi. Kemudian, route ini mengarahkan ke tampilan ' notif_eksplorasi.blade.php', yang merupakan file view untuk menampilkan halaman beserta notifikasi.

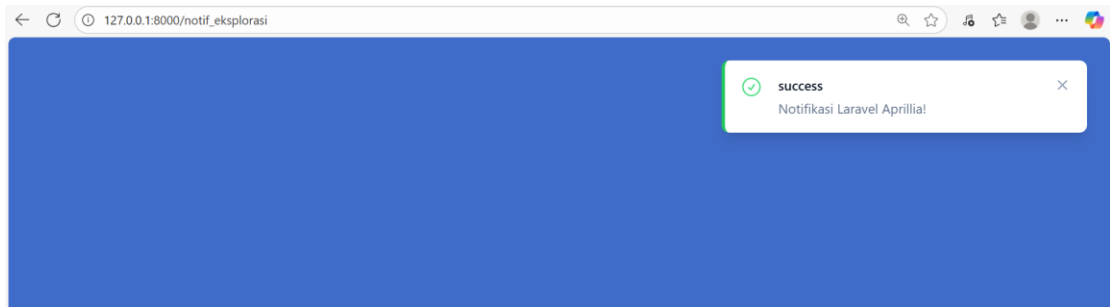
8.



Untuk membuktikan integrasi notifikasi pada laravel sudah berhasil atau tidak, kita jalankan servernya terlebih dahulu di terminal Laragon dengan perintah 'php artisan serve'

Selanjutnya salin URL nya dan tambahkan nama route yang tadi dibuat sebagai contoh ‘/notif_eksplorasi’

9.



Pada URL ‘127.0.0.1:8000/notif_eksplorasi’, notifikasi dengan status success muncul di sisi kanan atas layar dengan pesan “Notifikasi Laravel Aprillia!”. Ini menandakan integrasi paket Laravel Notify sudah berjalan dengan baik dan dapat menampilkan pesan ke pengguna secara visual.