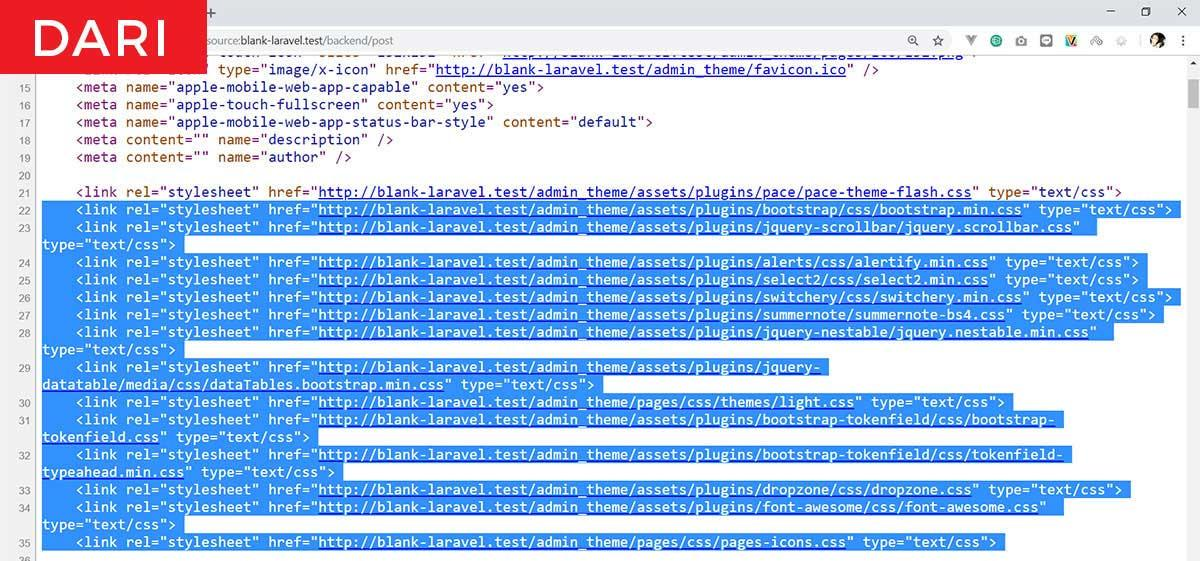
**Manajemen Asset Frontend Website jadi Lebih Mudah dengan Laravel Mix**

Laravel adalah salah satu framework PHP yang cukup terkenal saat ini karena lengkap dan mudah digunakan. Sangat banyak tools yang disiapkan yang kalau digunakan bisa sangat bermanfaat lho.. Salah satunya adalah Laravel Mix.

Sebelum membahas Laravel Mix lebih lanjut, kira-kira hasil akhir yang diharapkan dari menggunakan Mix adalah seperti ini :





Cukup menarik bukan? Laravel Mix adalah package frontend yang disiapkan untuk manajemen asset di laravel. Asset yang dimaksud disini bisa berupa file CSS, JS, image, font, dan sebagainya..

Manajemen yang dimaksud itu misalnya seperti *minify script* untuk mengecilkan ukuran dokumen CSS/JS, *combining script* untuk menggabungkan beberapa file CSS/JS menjadi satu, *script versioning*untuk merecord versi asset sehingga ketika ada perubahan nggak akan ngecache di browser client. Dari 3 kegunaan itu saja menurut saya sudah sangat efektif banget, terutama script combining dan versioningnya..

Untuk menggunakan Laravel Mix, kita harus sudah menginstall npm dulu di server / local kita. Jalankan command "npm install" untuk menginstall dependencynya terlebih dahulu. Jika sudah selesai, kita buka file webpack.mix.js yang ada di folder project Laravel kita. Isi awalnya kira-kira seperti ini :

const mix = require('laravel-mix');

/\*

|--------------------------------------------------------------------------

| Mix Asset Management

|--------------------------------------------------------------------------

|

| Mix provides a clean, fluent API for defining some Webpack build steps

| for your Laravel application. By default, we are compiling the Sass

| file for the application as well as bundling up all the JS files.

|

\*/

mix.js('resources/js/app.js', 'public/js')

.sass('resources/sass/app.scss', 'public/css');

Selain baris pertama, sisanya boleh dihapus kok.. Itu hanya script contoh yang diberikan dari Laravel.

File template dasar umumnya memanggil banyak file CSS/JS, dan secara pagespeed score pun ini nggak terlalu baik karena HTTP requestnya jadi terlalu banyak. Sebagai contoh, bisa lihat screenshot coding dibawah ini :

CSSnya memiliki 5 file yang dipanggil dengan 5 tag yang berbeda-beda. Kita bisa menggabungkan 5 file CSS tersebut menjadi 1 file dengan perintah mix.style() di webpack. Scriptnya adalah seperti ini :

mix.styles([

'public/template/css/bootstrap.min.css',

'public/template/css/style.css',

'public/template/css/font-awesome.css',

'public/template/css/icon-font.min.css',

'public/template/css/animate.css',

],

'public/css/hasil\_combine.css').version();

Hal yang sama juga bisa dilakukan untuk file JS.

mix.scripts([

'public/template/js/jquery-1.10.2.min.js',

'public/template/js/Chart.js',

'public/template/js/wow.min.js',

'public/template/js/jquery.nicescroll.js',

'public/template/scripts.js',

'public/template/js/bootstrap.min.js',

],

'public/js/hasil\_combine.js').version();

Yang perlu diperhatikan adalah,, mix.styles() untuk CSS, dan mix.scripts() untuk JS. Cara kerja kedua fungsi ini sama, yaitu parameter pertama berisi array list lokasi file2 apa saja yang ingin digabungkan, dan parameter kedua berisi lokasi file penyimpanan hasil combine. Method tambahan version() sifatnya optional untuk membantu proses file versioning saja.

Jika sudah dibuat, kita tinggal menjalankan command "npm run production" jika ingin hasil script yang terminify, atau "npm run development" jika hanya ingin menggabungkan tanpa minify. Jika sudah dijalankan dan sukses, maka hasilnya adalah seperti ini :

Kalau ada error, kemungkinan ada masalah di sintaks titik koma dan semacamnya, jadi diharapkan dibuat dengan teliti ya..

File hasil combine tersebut sekarang sudah bisa diakses di lokasi yang kita definisikan. Sekarang kita tinggal memanggil file tersebut di view laravel menggantikan yang lama..

Voila, sekarang file CSS dan JS sudah sukses dicombine.. Memang sih, script combine ini nggak terlalu cocok digunakan untuk semua plugin, terutama CSS/JS yang banyak mengakses ke direktori lain secara relative path (biasanya pas udah running akan ada error 404 kalau ada plugin yang seperti itu). Tapi teknik ini akan sangat membantu meringankan website yang menggunakan banyak CSS dan JS dalam 1 dokumen.