

Laravel

DOCUMENTATION



Muhammad Dandy Chrisnandy
XI RPL 2

Cara Install Framework Laravel PHP di XAMPP

Framework laravel ini berbeda sekali dengan framework yang lainnya khususnya pada tahap penginstalan, itulah mengapa saya akan menjelaskan kepada anda sekalian.

Berikut ini langkah-langkah untuk instalasi framework laravel di XAMPP :

Sebelum melakukan penginstalan, rekan-rekan harus memenuhi syarat awal untuk dapat melakukan instalasi. syarat awal tersebut merupakan software yang harus ada. Kalau tidak, maka anda gagal dalam memlakukan instalasi tersebut.

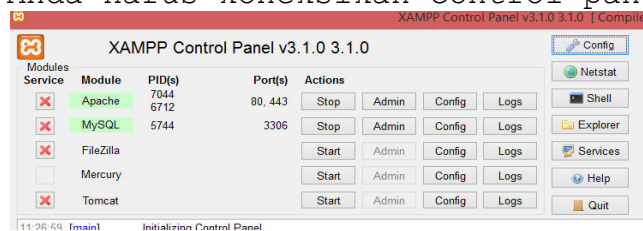
1. Software yang dibutuhkan
 - o Versi PHP minimal versi 4.3.7
 - o Composer
 - o Tentunya Laravel (Syarat Akhir)
2. Langkah-langkah Install
 - o Enable OpenSSL
 - o Install Composer
 - o Install Laravel
 - o Installation Complete

Berikut Cara Cara nya :

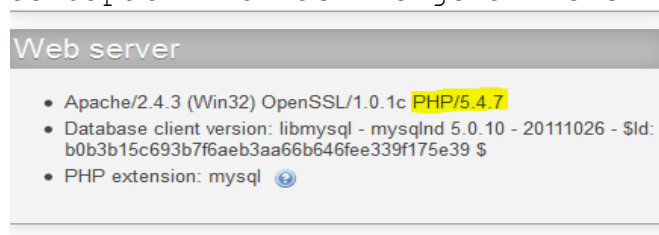
(Mengetahui Versi PHP)

Untuk mengetahui versi PHP, anda harus cek versi nya di phpmyadmin. caranya adalah :

- Anda harus koneksikan control panel XAMPP nya.



- Buka browser dan ketikkan `localhost/phpmyadmin` sama seperti biasanya. Lalu lihat ke arah kanan terdapat sidebar dengan judul Web Server, disiti terdapat informasi mengenai versi PHP anda.



(Download Composer)

Software Composer ini penting untuk installasi laravel, karena dalam software composer terdapat paket library yang dapat anda manfaatkan dalam pembuatan project menggunakan framework laravel.

Untuk Download Composer, bisa di check di website resminya yaitu : <https://getcomposer.org/download/>

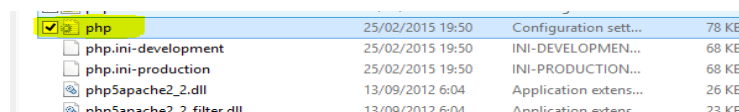
Setelah rekan-rekan memenuhi syarat install laravel versi php minimal 5.3.7 dan sudah download software composer maka selanjutnya rekan-rekan download framework laravel nya. Namun ternyata framework laravel dapat di download ketika sudah install composer nya.

Berikut Langkah-Langkah Untuk Instal Laravel :

- Enable kan OpenSSL

OpenSSL terdapat di file php.ini posisinya pada drive default yaitu (C:\xampp\php\php.ini).

Php.ini biasanya dalam folder php tersebut ada 3 macam, yaitu : php.ini, php.ini-development, dan php.ini-production. Yang harus anda edit OpenSSL nya adalah file php yang bertanda kuning dibawah ini.



<input checked="" type="checkbox"/> php	25/02/2015 19:50	Configuration sett...	78 KB
<input type="checkbox"/> php.ini-development	25/02/2015 19:50	INI-DEVELOPMEN...	68 KB
<input type="checkbox"/> php.ini-production	25/02/2015 19:50	INI-PRODUCTION...	68 KB
<input type="checkbox"/> php5apache2_2.dll	13/09/2012 6:04	Application extens...	26 KB
<input type="checkbox"/> php5apache2_2_filter.dll	13/09/2012 6:04	Application extens...	23 KB

Untuk membuat OpenSSL enable adalah anda harus cari di file tersebut tulisan extension=php_openssl. Lalu ubah OpenSSL dengan cara menghapus komentar nya, sehingga menjadi seperti dibawah ini :

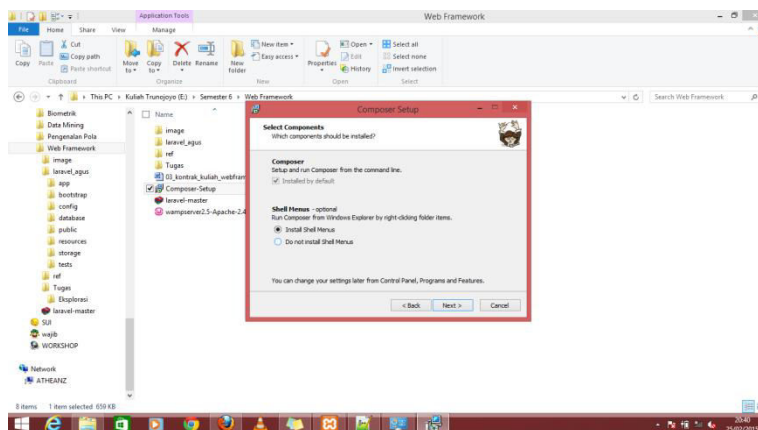
```
;extension=php_openssl.dll menjadi  
extension=php_openssl.dll
```

Fungsi dari enable ini adalah membuat software composer yang akan anda install di langkah selanjutnya dapat berjalan di cmd.

- **Proses install composer**

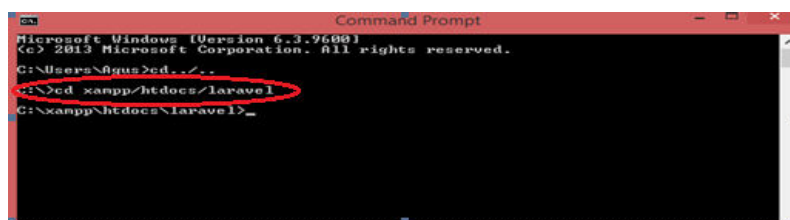
Cara install aplikasi ini sama seperti aplikasi yang biasa anda install di laptop masing-masing, nanti anda akan di hadapkan pada pilihan install shell menus dan do not install shell menus.

Anda pilih install shell menus saja, karena dengan begitu Anda dapat langsung membuka composer di cmd langsung dengan cara klik kanan pada folder yang akan di install laravel.



- **Install laravel**

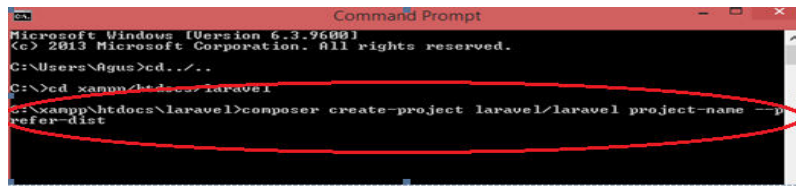
Instalasi laravel dilaksanakan dengan memanfaatkan command prompt windows anda. Sebelum membuka cmd anda, **buatlah terlebih dahulu folder** dengan nama laravel di path C:\xampp\htdocs\ . Kemudian buka cmd anda lalu **masuklah ke path folder laravel** yang anda buat tadi, contoh :



Lalu install framework laravel nya dengan meletakkan sintak dibawah ini pada path yang sudah diketikkan di cmd diatas,

composer create-project laravel/laravel project-name --prefer-dist

Gantilah **project-name** sesuai dengan nama project yang anda inginkan.



```
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

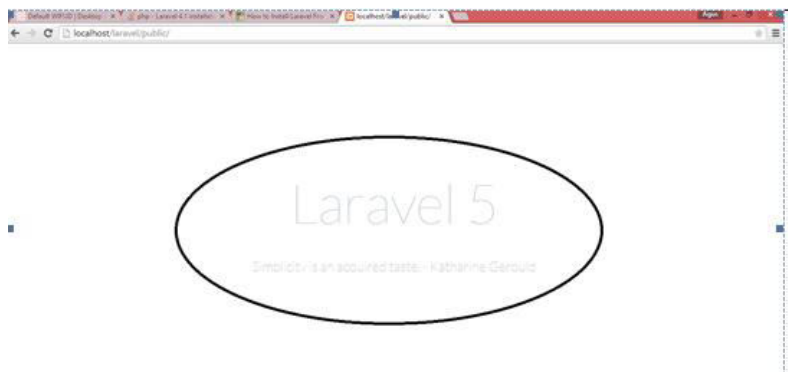
C:\Users\Agus>cd ../..
C:\>cd xampp\htdocs\laravel
C:\xampp\htdocs\laravel>composer create-project laravel/laravel project-name --prefer-dist
```

Proses untuk install laravel membutuhkan waktu yang cukup lama, jadi untuk install laravel ini memang membutuhkan kesabaran yang berbeda. *"Tunggu sampai Laravel terinstall sampai lengkap"*.

Setelah instalasi complete, untuk memastikan framework laravel terinstall dengan benar di laptop/komputer rekan-rekan, silahkan cek di folder yang rekan-rekan install laravel tadi. Di local disk

C:\xampp\htdocs\nama-folder. Atau cek di browser rekan-rekan dengan url **localhost/nama-folder/public**. Karena memang file **index.php** nya terdapat di folder public.

Sehingga akan muncul tampilan seperti dibawah ini :



Membuat Halaman Web Pertama Dengan Laravel

Router

Kebanyakan framework PHP melakukan mapping otomatis antara URL dengan controller. Oleh sebab itu, untuk membuat sebuah halaman, Anda minimal harus membuat satu buah controller. Tetapi, laravel melakukan pendekatan yang sedikit berbeda. Mapping URL harus didaftarkan secara manual di file `app/routes.php`.

Anda bisa me-mapping suatu URL ke sebuah controller, tapi hal tersebut tidak wajib. Untuk sebuah aksi sederhana, Anda bisa langsung melakukannya di Router.

mari kita buat sebuah halaman selamat datang dengan URL [http://localhost/laravel/public/halo] yang akan menampilkan tulisan "Halo world".]

Buka file app/routes.php, lalu tambahkan kode untuk routing:

```
Route::get('/halo', function()
{
    return "Halo, World";
});
```

Selesai, buka browser Anda dan ketikkan URL [http://localhost/laravel/public/halo], Anda sudah berhasil membuat halaman web pertama dengan laravel.

Controller

Dalam konsep MVC biasanya sebuah URL dipetakan ke sebuah controller, dan karena konsep MVC sudah mendarah daging di kalangan web programmer, maka kita akan membuat satu halaman lagi dengan url **/halo-juga**, outputnya mirip, tapi kali ini menggunakan controller.

Buka kembali file app/routes.php, lalu tambahkan router baru:

```
Route::get('/halo-juga', 'SiteController@haloJuga');
```

Penjelasan dari kode di atas, jika ada yang meminta url **/halo-juga**, maka laravel akan mengeksekusi fungsi haloJuga() di dalam SiteController. Nah, karena SiteController-nya belum ada, maka langkah selanjutnya adalah membuat controller.

Tambahkan file baru app/controllers/SiteController.php:

```
class SiteController extends BaseController {
    public function haloJuga()
    {
        return 'halo juga, bro';
    }
}
```

Buka [<http://localhost/laravel/public/halo-juga>] dan lihat hasilnya. Selamat, Anda berhasil membuat halaman kedua dengan laravel, kali ini memanfaatkan controller.

View

Pada contoh di atas, kita cuma menampilkan string sederhana ke browser. Ngomong-ngomong soal browser, agar kelihatan valid tentunya string tersebut harus dibungkus dengan tag html yang lengkap, seperti berikut ini:

```
<html>
  <head>
    <title>d</title>
  </head>
  <body>
    halo juga, bro
  </body>
</html>
```

Tugas Anda selanjutnya adalah memodifikasi contoh yang sudah diberikan, sehingga bisa menghasilkan tag html yang lengkap.

Hehe, apakah Anda akan memodifikasi kodenya menjadi seperti ini:

```
public function haloJuga()
{
    return '<html><head><title></title></head><body>halo juga, bro</body></html>';
}
```

Memformat tampilan langsung dari controller jelas tidak menyenangkan, dan memang tidak dianjurkan. Tampilan adalah tugas front-end developer, dan front-end developer tidak suka controller, mereka cuma suka view. Untuk menghindari kebencian mereka, mari kita buat view-nya.

Modifikasi kembali fungsi `haloJuga()` seperti berikut ini:

```
public function haloJuga()
{
    return View::make('halo_juga');
}
```

Fungsi `View::make('nama_file')` akan memanggil file view terpisah sesuai nama yang diberikan. Selanjutnya kita buat file baru `app/views/halo_juga.php`:

```
<html>
  <head>
    <title></title>
  </head>
  <body>
    halo juga, bro
  </body>
</html>
```

Jika Anda ingin menerapkan prinsip satu folder view untuk satu controller, yang berarti file `app/view/halo_juga.php` dipindahkan ke `app/views/site/halo_juga.php`, maka kodenya tinggal diedit sedikit menjadi `View::make('site.halo_juga')`. Tanda titik (dot) digunakan sebagai separator folder.