

## Tools untuk develop Web 2 (Laravel)

### A. Instalasi laravel

1. Install XAMPP pada windows, gunakan XAMPP versi 7.2.0 yang berisi PHP versi 7.1.3 keatas.
2. Setelah menginstall XAMPP, maka selanjutnya perlu menginstall Composer.  
*Composer adalah dependency management atau bisa juga dibidang tool untuk mengelola paket(Package Manager/Management) yang terkait dengan PHP dan tentu untuk Laravel juga. Download Composer di website <https://getcomposer.org>.*
3. Setelah PHP dan Composer terinstall, selanjutnya install Laravel di windows, pastikan untuk masuk ke folder c:/xampp/htdocs (sesuaikan dengan direktori instalasi xampp) melalui Command Prompt (CMD) dengan mengetik perintah :

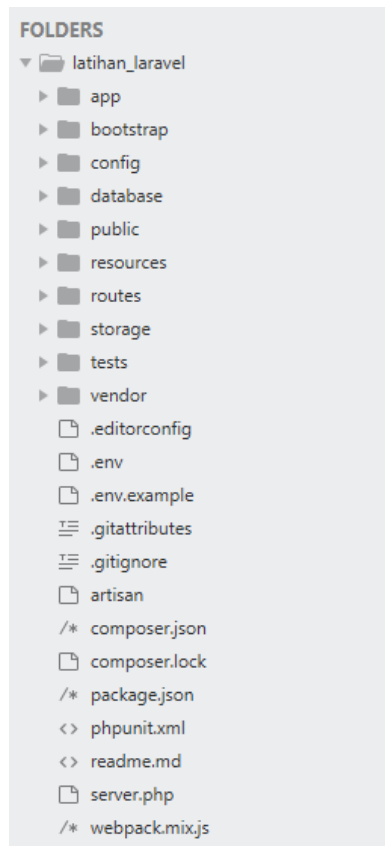
```
cd c:\xampp\htdocs
```

4. Sebelum menginstall Laravel, pastikan kebutuhan (requirements) servernya terpenuhi, antara lain sebagai berikut :
  - PHP >= 7.1.3
  - OpenSSL PHP Extension
  - PDO PHP Extension
  - Mbstring PHP Extension
  - Tokenizer PHP Extension
  - XML PHP Extension
  - Ctype PHP Extension
  - JSON PHP Extension
  - BCMath PHP Extension
5. Lalu jalankan perintah dibawah ini untuk menginstall laravel.

```
composer create-project laravel/laravel latihan_laravel "5.7.*"
```

*Perintah tersebut akan membuat sebuah folder baru dengan nama latihan\_laravel di c:\xampp\htdocs. Ketika menjalankan perintah tersebut maka secara otomatis composer akan mendownload semua paket dependencies yang dibutuhkan untuk menjalankan Laravel ke folder latihan\_laravel, dan perlu menunggu hingga selesai serta pastikan koneksi internet lancar.*

6. Jika sudah selesai maka bisa dilihat folder latihan\_laravel yang sudah berisi struktur file sebagai berikut.



7. Untuk menjalankan laravel maka masuk ke folder latihan\_laravel dengan Command Prompt (CMD), lalu jalankan perintah artisan berikut:

```
php artisan serve
```

8. Sekarang akses halaman web Laravel di browser dengan mengakses <http://localhost:8000>

## B. Routing

- Laravel memiliki sebuah fitur yang digunakan untuk mendaftarkan semua URI yang bisa diakses oleh pengguna aplikasi berdasarkan respon dari HTTP. Fitur inilah yang dinamakan route.
- Routing sendiri adalah istilah untuk memberikan alur, atau arahan melalui URL pada framework laravel. Routing sendiri memang salah satu konsep yang sudah di adaptasi pada semua framework php dengan tujuan untuk mengelola url yang kita buat agar bisa berfungsi

maksimal, kegunaan routing sendiri memang tidak diragukan lagi, baik itu untuk keperluan enkripsi, parameter, ataupun kita bisa juga membuat url agar memiliki sifat SEO Friendly. Berikut langkah untuk menerapkan routing.

1. Masuk ke dalam folder routes/web.php
2. Tambahkan script berikut :

```
Route::get('/page1', function () {  
    return view('page1');  
});
```

*Kode diatas, berfungsi agar saat url /page1 di akses, maka akan tampil view page1.*

3. Setelah menambahkan kode diatas, masuk ke dalam folder, resources/views, kemudian buat sebuah file baru dengan nama page1.blade.php.
4. Setelah membuat file tersebut, isi halaman tersebut dengan kode html seperti berikut :

```
<!DOCTYPE html>  
<html>  
<head>  
    <title>Page 1</title>  
</head>  
<body>  
<p>Belajar Laravel</p>  
</body>  
</html>
```

5. Kemudian silahkan akses di browser dengan url berikut : <http://localhost:8000/page1>
- Implementasi routes juga bisa dilakukan untuk menjalankan sebuah function di Controller. Jadi, semua proses akan dilakukan di Controller sesuai dengan kaedah MVC itu sendiri. Cara mengarahkan route laravel ke controller, gunakan script berikut yang juga di tambahkan di folder routes/web.php :

```
Route::get('post/{id}', 'PostController@detail');
```

- Jika mengakses url [http://localhost:8000/post/\(ID bebas\)](http://localhost:8000/post/(ID bebas)) maka prosesnya akan diteruskan ke function detail yang berada pada PostController.

### C. Controller

- Laravel memiliki yang namanya controller untuk pengolah data. Sama seperti controller pada framework PHP lainnya, karena sama-sama menerapkan konsep MVC dalam pengembangan dan penggunaan framework PHP nya.
- Controller merupakan penghubung antara view dan model. Jadi secara mudahnya, controller bisa dipahami sebagai pengatur view dan model.
- Mungkin di beberapa contoh kasus, memerlukan penerapan logika atau pengolahan data, maka controller lah yang berperan, baru kemudian ditampilkan ke bagian view (user interface) aplikasi.
- Controller Laravel berada di dalam folder app/Http/Controllers/
- Ada 2 cara untuk membuat controller pada Laravel.
- Cara pertama membuat controller Laravel adalah sebagai berikut :  
Buat file baru pada direktori app/Http/Controllers/ , kemudian beri nama MahasiswaController. Lalu isi controller tersebut dengan kode dibawah ini :

```
<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

class MahasiswaController extends Controller
{
    //di sini isi controller mahasiswa
}
```

Perhatikan pada kode diatas, saat mendeklarasikan class controller, wajib meng-extends controller dari Laravel. Kemudian perhatikan dari penulisan class nya, wajib menuliskan sesuai dengan nama file nya.

- Cara kedua membuat controller adalah dengan menggunakan php artisan.
- Php artisan adalah fitur unggulan yang ada pada Laravel, yang dibuat untuk memudahkan kita dalam pengembangan menggunakan Laravel.
- Dengan menggunakan php artisan, kita dapat membuat controller secara otomatis menggunakan perintah php artisan. Berikut ini adalah langkah yang harus dilakukan :
  1. Buka terminal atau command prompt, kemudian arahkan ke direktori project Laravel.
  2. Setelah masuk ke direktori latihan\_laravel, ketik script berikut :

```
php artisan make:controller DosenController
```

Perintah `make:controller` adalah perintah php artisan untuk membuat controller dan DosenController adalah nama controller yang dibuat.

3. Jika berhasil, maka akan muncul pesan : Controller created successfully.
- Selanjutnya untuk contoh penggunaan controller, dapat dilakukan langkah dibawah ini :
  1. Buat route baru, lihat langkah pada routing, kemudian tambahkan kode berikut.

```
Route::get('dosen', 'DosenController@index');
```

2. Buka file controller DosenController.php
3. Pada function index, tambahkan kode berikut :

```
public function index(){  
    return "Latihan membuat method index, pada  
        controller dosen";  
}
```

4. Sekarang, coba untuk mengakses url : <http://localhost:8000/dosen>

#### D. Passing Data Controller Ke View Laravel

- Passing data adalah proses mengoper data. Data yang dioper adalah data yang ada pada controller ke view untuk ditampilkan. Untuk melakukan passing dapat menggunakan fungsi `view()`
- Fungsi `view()` digunakan untuk memanggil view yang ingin ditampilkan.
- Berikut adalah langkah untuk memanggil view dari controller :
  1. Buka controller DosenController, pada `latihan_laravel/app/Http/Controllers/DosenControllers.php`
  2. Buat view baru dengan nama `dosen.blade.php`. dan isikan dengan kode berikut.

```
<!DOCTYPE html>  
<html>  
<head>  
    <title>Page 2</title>  
</head>  
<body>  
<p>Ini adalah tampilan data dosen</p>  
</body>  
</html>
```

3. Selanjutnya pada method index dalam controller DosenController, panggil `dosen.blade.php` dengan mengubah script sebelumnya menjadi seperti ini :

```
public function index(){
    return view('dosen');
}
```

4. Selanjutnya akses url : <http://localhost:8000/dosen>
- Untuk memarsing data dari controller ke view, bisa mengirim pada fungsi view(), sebagai contoh, ikuti langkah berikut :
  1. Buka lagi method index() pada controller DosenController.
  2. Ubah method index, menjadi seperti berikut :

```
public function index(){
    $nama = "Henny Alfianti";
    return view('dosen', ['nama' => $nama]);
}
```

3. Lalu, edit view dosen.blade.php, menjadi seperti berikut :

```
<!DOCTYPE html>
<html>
<head>
    <title>Page 2</title>
</head>
<body>
<p>Tutorial Laravel</p>
<p>Nama : {{$nama}}</p>
</body>
</html>
```

4. Kemudian jalankan pada browser.

## E. View (Blade Template)

- Laravel membuat ekstensi sendiri untuk menangani bagian view, yaitu ekstensi blade.php.
- Pada beberapa penjelasan sebelumnya, telah dibahas bagaimana cara membuat view menggunakan blade.php.
- Pada pembahasan ini, kita akan mempelajari tentang sistem templating pada Laravel.
- Berikut adalah langkah untuk menerapkan sistem templating :
  1. Buat 3 buah route pada latihan\_laravel/routes/web.php, dengan kode sbb :

```
Route::get('/blog', 'BlogController@home');
Route::get('/blog/tentang', 'BlogController@tentang');
Route::get('/blog/kontak', 'BlogController@kontak');
```

2. Lalu buatlah controller dengan nama BlogController, menggunakan php artisan seperti berikut: `php artisan make:controller BlogController`
3. Setelah selesai, buatlah 3 buah method didalam controller tersebt, dengan script sebagai berikut :

```
class BlogController extends Controller
{
    public function home(){
        return view('home');
    }

    public function tentang(){
        return view('tentang');
    }

    public function kontak(){
        return view('kontak');
    }
}
```

4. Kemudian buat template masternya terlebih dahulu, template master disini adalah template induk nya. Buat sebuah view baru dengan nama master.blade.php

```

<!DOCTYPE html>
<html>
<head>
  <title>Latihan Templating</title>
  <link rel="stylesheet" type="text/css" href="{{ asset('/css/app.css') }}">
</head>
<body>
  <header>
    <h2>Blog TEDC</h2>
    <nav>
      <a href="/blog">HOME</a>
      |
      <a href="/blog/tentang">TENTANG</a>
      |
      <a href="/blog/kontak">KONTAK</a>
    </nav>
  </header>
  <hr/>
  <br/>
  <br/>

  <!-- bagian judul halaman blog -->
  <h3> @yield('judul_halaman') </h3>

  <!-- bagian konten blog -->
  @yield('konten')

  <br/>
  <br/>
  <hr/>
  <footer>
    <p>&copy; <a href="#">www.poltekdedc.ac.id</a> 2018 - 2019</p>
  </footer>

</body>
</html>

```

5. Buat 3 buah view baru, yaitu view home, tentang dan kontak.

6. Script untuk view home :

```

<!-- Menghubungkan dengan view template master -->
@extends('master')

<!-- isi bagian judul halaman -->
@section('judul_halaman', 'Halaman Home')

<!-- isi bagian konten -->
@section('konten')
  <p>Ini Adalah Halaman Home</p>
@endsection

```

7. Script untuk view tentang :



```

<!-- Menghubungkan dengan view template master -->
@extends('master')

<!-- isi bagian judul halaman -->
@section('judul_halaman', 'Halaman Tentang')

<!-- isi bagian konten -->
@section('konten')

    <p>Ini Adalah Halaman Tentang</p>
    <p>
        Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod
        tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam,
        quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo
        consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse
        cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non
        proident, sunt in culpa qui officia deserunt mollit anim id est laborum.
    </p>

@endsection

```

8. Script untuk view kontak :

```

<!-- Menghubungkan dengan view template master -->
@extends('master')

<!-- isi bagian judul halaman -->
@section('judul_halaman', 'Halaman Kontak')

<!-- isi bagian konten -->
@section('konten')

    <p>Ini Adalah Halaman Kontak</p>

    <table border="1">
        <tr>
            <td>Email</td>
            <td>:</td>
            <td>henny.alfianti@gmail.com</td>
        </tr>
        <tr>
            <td>Phone</td>
            <td>:</td>
            <td>0812-3456-7890</td>
        </tr>
    </table>

@endsection

```

9. Kemudian jalankan dengan url : <http://localhost:8000/blog>

## F. Bootstrap (CSS framework)

- Pada project laravel, sebenarnya sudah memiliki file css bootstrap secara default pada pertama kali kita menginstall laravel.
- Letaknya ada pada file app.css dalam folder css.

- Untuk menggunakannya dapat menambahkan file css <link rel> pada halaman view, untuk dapat menggunakannya. Berikut scriptnya :

```
<title>Latihan Templating</title>
<link rel="stylesheet" type="text/css" href="{{ asset('/css/app.css') }}">
```

- G. Exercise: Buatlah 5 halaman yang telah terhubung seperti pada sistem templating yang udah digabungkan dengan bootstrap.