

# Long-Life Learning: Web-Based Development with Laravel Platform

Gaudensius Dimas Prasetyo Suprpto - 13514059  
*Program Studi Teknik Informatika*  
*Sekolah Teknik Elektro dan Informatika*  
*Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia*  
<sup>1</sup>13514059@std.stei.itb.ac.id

**Abstract**—Web-based development is an activity done to develop or build a website, whether in front-end manner, back-end manner, or webpage-designing manner.

**Keywords**—Internet, Web-Based Development, Laravel.

## I. BACKGROUND

Internet is the technology that human being has created to connect people around the world at every side of the world to each others. This usage of internet has led human being to yet again another new era of humankind, the interconnected era.

The widely usage of this technology extremely encourage the transparency of data, where almost all informations around the world are accessible for anyone, anytime, and anywhere. The informations can be shown in very numerous manners, and one of the most widely-used manner is to show them in websites or pages, which is created by web-based applications.

## II. INTRODUCTION

Web-based development is an activity that involves whether developing an existing web-based application, or building a new one. These web-based application are not only those which are hosted widely to the public internet, but also those hosted at the local intranet. These applications are made by those known as the web developers, whether front-end developer, back-end developer, or web designer.

Web-based development itself has very many ways to develop, and very many programming languages and platforms to work with. The most basic programming language used to build a web application is the client-sided HTML (Hyper Text Markup Language), as most of web browsers has the ability to render HTML into web pages that we can see. As for the most common server-sided programming language we have PHP (Hyper-text PreProcessor).

What this paper is going to be is about learning Laravel, a PHP-based platform for web development.

## III. LARAVEL AS WEB DEVELOPMENT PLATFORM

### A. Installing Laravel<sup>[1]</sup>

Before going about the insides of Laravel, it is crucial to study how to install Laravel first. Laravel has a server requirement as follows:

- PHP version 5.6.4
- OpenSSL PHP Extension
- PDO PHP Extension
- Mbstring PHP Extension
- Tokenizer PHP Extension
- XML PHP Extension

Installing Laravel requires Composer installed, as Laravel utilizes Composer as its dependencies manager. To install Laravel to the machine, execute the following command at command prompt:

```
composer global require "laravel/installer"
```

After installing Laravel to the machine, creating a new Laravel project is as simple as executing the following command at desired directory:

```
laravel new AppName
```

or using Composer's Create-Project...

```
composer create-project --prefer-dist  
laravel/laravel AppName
```

Assuming that PHP is installed locally, serving the application using PHP's built-in development server can be done by using `serve` Artisan command:

```
php artisan serve
```

Note that this command will start the server at address `http://localhost:8000`.

## B. Configuring Laravel <sup>[1]</sup>

After installing, there are several things needed to configure, such as follows:

- Public directory, as the root of web should be public.
- Configuration files, located at the config directory of the project.
- Directory permissions, as the storage and bootstrap/cache needs to be writable by the server in order to run Laravel.
- Application key, done by running command `php artisan key:generate` at the project's root directory.
- Environment file (`.env`), located at the project's root, if it does not exist, it must be copied from the `.env.example` at the same directory.

## C. Laravel Basic <sup>[1]</sup>

After installing and configuring Laravel, the main basics of Laravel now can be learned. The basics of Laravel are as follows:

- Routing  
Routing can be done by editing the routes at `routes/web.php` file. The router methods available to use are:

```
Route::get($uri, $callback);
Route::post($uri, $callback);
Route::put($uri, $callback);
Route::patch($uri, $callback);
Route::delete($uri, $callback);
Route::options($uri, $callback);
```

Note that we can also make logic operations at the `routes/web.php` file, such as processing the `get` parameter before going to the page and other things.

- Controller  
Controllers are used to simplify the logic operations done at `routes/web.php` file. For example, the Controller `Example.php` located at `App\Http\Controllers` is as follows:

```
<?php
namespace App\Http\Controllers;

use App\User;
use App\Http\Controllers\Controller;

class Example extends Controller
{
    public function show()
    {
        return "Hello World";
    }
}
```

Then, we can define a route to use the controller's existing function at the `routes/web.php` with for example:

```
Route::get('hello', 'Example@show')
```

Then, when `http://localhost:8000/hello` is accessed through web browser, it will route into the `show` function that the "Example" Controller and run it, and thus showing the text "Hello World".

- Views  
Views contain the HTML served by the application. This enables to separate the application logic from presentation logic, or as we know, to separate the back-end (Controllers) from the front-end (Views). Views are located at the `resources/views` directory. For example, a `greeting.blade.php` is as follows:

```
<html>
  <body>
    <h1>Hello, {{$name}}</h1>
  </body>
</html>
```

Then, a route is defined as follows:

```
Route::get('greet', function () {
    return view('greeting', [
        'name'=>'James'
    ]);
});
```

Then, when `http://localhost:8000/greet` is accessed through web browser, it will search the view at the `resources/views` directory and show the corresponding view, the `greeting.blade.php` file.

Also note that the PHP parameter `$name` is passed by the second parameter of `view` function. For clearer explanation, another example will be shown :

```
return view('greeting', [
    'name'=>'James',
    'nim'=>'13514999'
]);
```

This will pass two PHP parameters, where `$name` is "James", and `$nim` is "13514999"

#### IV. CONCLUSION

For web development means, Laravel is a considerable platform because its basic features can be developed into a bigger feature by our own

#### V. ACKNOWLEDGMENT

All my deepest gratitude to Miss Ayu as the lecturer of IF 3280 – SocioInformatics and Professionalism for giving such assignment to encourage Long Life Learning and make it into practice.

Also my special thanks for all of my family, my friends, and other people who have supported me so I can finish this paper.

#### REFERENCES

- [1] <https://laravel.com/docs/5.4/>, 20:00, 5 May 2017
- [2] [https://www.ieee.org/conferences\\_events/conferences/publishing/templates.html](https://www.ieee.org/conferences_events/conferences/publishing/templates.html), 19:45, 5 May 2017

#### STATEMENT

Hereby I declare that this paper is my own writing, and not a form of plagiarism, nor a translated paper

Bandung, 5 May 2017

A handwritten signature in black ink, appearing to read 'Dimas', with a stylized underline.

Gaudensius Dimas Prasetyo Suprpto / 13514059