

Web development with laravel

Ahmad Faiq Rahman – 13514081
Computer Science / Informatics
School of Electrical Engineering and Informatics
Bandung Institute of Technology, Jl. Ganesha 10 Bandung 40132, Indonesia
13514081@std.stei.itb.ac.id

Abstract—In this paper I would like to share about the things that I learned to be able to develop a web application with a tool named laravel. Slight introduction about the tool and what I do in order to use it in web development.

Keywords—*laravel; web development; web app; development tool;*

I. INTRODUCTION

At the first week of this course, we were asked to write some tools that we'd like to learn this semester in order to fill up our bars regarding the technical skills of an informatics bachelor. I myself had written down that I wanted to learn about security, anything about security, that an informatics bachelor need to now so that I can become an expert at that thing. In my head, I was imagining to start learning about security on any sort of application, mobile, web, desktop, or any other platform one can code in.

As I said, those were my thoughts from the first days or weeks. When we haven't learned about all the assignments that was waiting to be touched by us during this semester. All those things that I mentioned, learning about software security in the middle of tasks and assignments that dues, are just left as a dream that is too good to be true. I wasn't only blaming the tasks and assignments, I'm also blaming myself for not being able to be enthusiastic enough to spare little of my time to learn about software security.

Enough about the daydream, here comes the reality. Since I haven't learned anything about software security, I will make this paper to talk about the assignment that I like the most in this semester, by like I mean I do this assignment with some more zeal than other assignments. Actually, by the time I write this paper, the project/assignments haven't been done yet, we are still in the middle of our development, me and my friends.

The project is actually a basic project, nothing unusual or anything, we were asked to make a website that allow interaction between a master's student and his/her preceptor during the making of the student's thesis and supervision from a higher position like dean to the thesis's progress. This website will be able to distinguish certain type of user and grant the user the ability to access all the necessary and appropriate information.

We were allowed to use any type of tools or anything in order to develop the web application. After a couple of trials and errors we decided to use laravel to develop this web application. That explains the title.

II. LARAVEL



Figure 1 Laravel Logo ([http://logos-download.com/wp-content/uploads/2016/09/Laravel logo wordmark logotype.png](http://logos-download.com/wp-content/uploads/2016/09/Laravel%20logo%20wordmark%20logotype.png))

Laravel is one of many web development tools that can be found and used across the globe. This tool, uses php language in order to make an engaging web. It is very easy to code, just like coding a normal php application. The only thing that was quite hard was the setup. Because of that, in this paper, I'd like to talk about some history and the setup of a laravel PHP framework.

A. History

Taylor Otwell, a .net developer in Arkansas, was using CodeIgniter when the first seeds of Laravel were planted.. "I couldn't add all the features I wanted," he says, "without mangling the internal code of the framework." He wanted something leaner, more simple, and more flexible.

Those desires, coupled with Taylor's .net background, spawned the framework that would become Laravel.

Traditionally, PHP had been like the wild west of programming languages. Because it is dynamically typed, there are a lot of things that could potentially go wrong with no warning. For example, in other languages, code is written in such a way that if a mistake is made it can be caught before code execution .Either by trying to use the code or by compiling it, the compiler would actually stop you, identify the problem in the code, and require you to fix it before moving forward.

Taylor aimed to solve all of those problems with Laravel. He used the ideas of the .net infrastructure, which Microsoft has built and spent hundreds of millions of dollars of research on.

With Laravel, Taylor sought to create a framework that would be known for its simplicity. He added to that simplicity an expressive syntax, clear structure, and incredibly thorough documentation. With that, Laravel was born.

He started with a simple routing layer, a really simple controller-type interface (model, view, controller or MVC); v1 and v2 were released in June 2011 and September 2011, respectively, just months apart. Laravel 3 came just over a year later, in February 2012, and this is when Laravel's user base and popularity really began to grow.

In May 2013, Laravel 4 was released as a complete rewrite of the framework, incorporated a package manager called Composer. Composer is an application-level package manager for PHP that allowed people to collaborate instead of compete.

Before Composer, there was no way to take two separate packages and use different pieces of those packages together to create a single solution. First you'd have to download the files, put them into a certain spot, figure out how to use them, and blah, blah, blah.

Composer became the tipping point where suddenly all of these framework authors starting looking at one another and started to see the benefits of collaboration instead of competition, and Laravel began to make a real impact in the PHP world. Today, we're using Laravel 5.2, with Taylor continuing to make improvements and innovations to the framework.

B. Setup

Laravel utilizes Composer to manage its dependencies. So, before using Laravel, you will need to make sure you have Composer installed on your machine.

- Via Laravel Installer

First, download the Laravel installer using Composer.

```
composer global require
"laravel/installer=~1.1"
```

Make sure to place the `~/composer/vendor/bin` directory in your PATH so the laravel executable can be located by your system.

Once installed, the simple `laravel new` command will create a fresh Laravel installation in the directory you specify. For instance, `laravel new blog` would create a directory named `blog` containing a fresh Laravel installation with all dependencies installed. This method of installation is much faster than installing via Composer:

```
laravel new blog
```

- Via Composer Create-Project

You may also install Laravel by issuing the Composer `create-project` command in your terminal:

```
composer create-project laravel/laravel
{directory} "5.0.*" --prefer-dist
```

Once installed, you should upgrade to the latest packages. First, remove `{directory}/vendor/compiled.php` file then change

your current directory to `{directory}` and issue `composer update` command.

- Scaffolding

Laravel ships with scaffolding for user registration and authentication. If you would like to remove this scaffolding, use the fresh Artisan command:

```
php artisan fresh
```

There are actually more steps needed to be done to setup laravel entirely. More information can be accessed through the reference I write in the end of this paper.

III. WHAT I LEARNED

After learning about the history and installation of laravel, we can now start the development of the website we demand. In this section, I'd like to share what I did in my development part.

I develop the dashboard of every kind of users that need to be implemented to the web application. In doing so, I need to learn about routing.

Routing is a some kind of controller that controls the route of the web application. Maybe it doesn't seem like it explained anything, but actually, it's that simple. This is one of the coolest features that is provided by laravel. This feature may allow the developer to list the blades/pages that can be accessed and which user could access it.

This feature makes page/blade access more organized and compact. With this feature developer might easily catch a picture of how the developed web would be look like when it is deployed.

IV. CLOSING

There are actually many more features that could be discussed like database migration, the composer itself, resources, etc. I would not go for any other feature because what I learned the most in this assignment/project is this one, routing feature.

I will explore more of those features in days to come as well as other tools that I have used this past semester. There are actually so many things that I have learned this semester, not only about the communication to computer (re: developing tools), but also about the communication with the user (re: groupmate).

This paper is just one of the things that I learned this semester. So many things that need to be explored, tried, and learned more in the future with this evolving era of technology. After all, as many people like to say, learning is a life-long process.

REFERENCES

- [1] <https://medium.com/vehikl-news/a-brief-history-of-laravel-5d55970885bc> (accessed May 5th 2017)
- [2] <https://laravel.com/docs/5.0> (accessed May 5th 2017)
- [3] <https://scotch.io/tutorials/simple-and-easy-laravel-routing> (accessed May 5th 2017)

ACKNOWLEDGMENT

This paper is made in order to finish an assignment for the respective course

IF3280 Socio-Informatika dan Profesionalisme

This paper is made by me and me alone with references that are written in this paper.

A handwritten signature in black ink, appearing to read 'Faiq', with a small circled 'R' above the final stroke.

Ahmad Faiq Rahman – 13514081