**LARAVEL FRAMEWORK**

**BY**

**AKAMSHU, GABRIEL EYUAH**

**ST/CS/HND/18/018**

**A SEMINAR**

**PRESENTED TO THE DEPARTMENT OF COMPUTER SCIENCE,**

**SCHOOL OF SCIENCE AND TECHNOLOGY,**

**FEDERAL POLYTECHNIC, MUBI, ADAMAWA STATE**

**IN PARTIAL FULFILMENT OF THE COURSE REQUIREMENTS FOR THE AWARD OF NATIONAL DIPLOMA (ND) IN COMPUTER SCIENCE**

**FEBRUARY, 2021**

**ABSTRACT**

*With the current explosion of Information Systems, the market offers a wide range of interesting technological solutions. Yet, this does not mean adopting a technology without considering its impact on the existing information system and user expectations. It is recommended to identify and implement the technological solutions most suited to the Information Systems strategy. Therefore, new methods are emerging and design tools are still evolving; the PHP Frameworks are part of it, which open up new perspectives in terms of information system enrichment. In this context, this seminar work focuses on the elaboration of Laravel, framework, which is becoming the most popular PHP frameworks and to learn and show the benefits of using Laravel framework to build web applications.*

**Introduction**

The implementation of an information system requires a study of existing programming environments in order to decide the best computational approach. Open source programming is very popular in the world of computing today. There are various open-source technologies currently in use-libraries, Frameworks, APIs, etc. The use of Frameworks stems from the need to quickly develop large applications, by promoting code reuse, testing and changes to an application (Christensson, 2019).

Framework is a product of collective intelligence, comprising many robust libraries and convenient tools from other developers. They help to reduce most of the repetitive tasks and complex tasks in simple interface, which means developers can write less and do more with highest quality in a certain amount of time. Therefore, using a reliable framework also help to lower the development cost. Furthermore, using a robust web framework also helps to strengthen the security of the application. It does not require developers to have a deep knowledge about security. There are many forms and types of cyber-attacks, fortunately, Laravel as well as most of web frameworks support several features to prevent basic security vulnerabilities such as SQL injection, cross-site request forgery (CRF) and cross-site scripting (XSS). Moreover, Laravel also provides a well-structured skeleton for building big projects. (Adam, 2019). The framework also supports modular architect which enables developers separate code into independent workable modules.

The most widely used programming language in web design is PHP due to its reliability, effective cost, and easy navigation that it can create highly interactive websites (Hardik, 2013). One of the best and superior frameworks for PHP compared to other frameworks is Laravel (James, 2019).

Laravel is designed to improve software quality, simplify authentication, ease routing, ease access, and increase the power within the website framework. Laravel is a framework application with elegant syntax and has broad functions such as security, password storage, reminders and reset passwords, encryption, and validation (Hendricks, 2019). According to a survey conducted in March 2015 concerning the popularity of the PHP framework, Laravel was the most popular framework, consecutively followed by Symfony2, Nette, CodeIgniter, Yii2, and others (James, 2019).

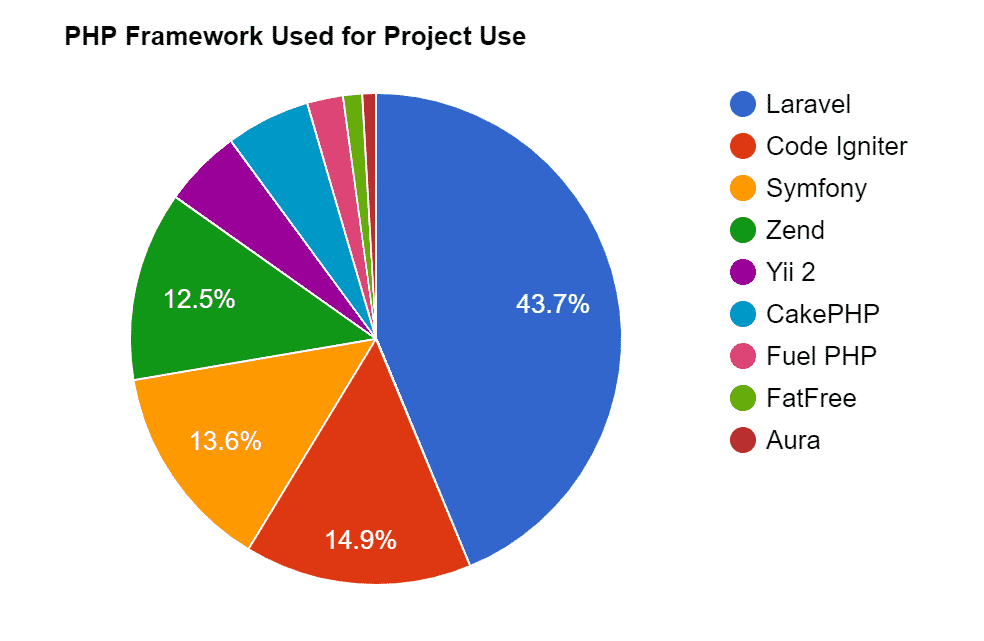


Figure 1: Framework Usage Statistics in the top 1 million websites (James, 2019).

# Laravel’s main features

This study will focus only on the features used during the building of the eCommerce web application, otherwise this work will not be large enough to cover the entirety of the features of the whole Laravel 4 framework.

## **Architecture**

Laravel is a web application framework that tries to ease the development process by simplifying repetitive tasks used in most of today’s web applications, including but not limited to routing, authentication, caching and sessions (Christensson, 2018). All the new Laravel projects come out of the box equipped with a full directory tree and also many placeholder files resulting in a structure permitting a quick start of the actual development process. This structure is nevertheless fully customizable. Here in the following figure is shown what such a structure looks like:

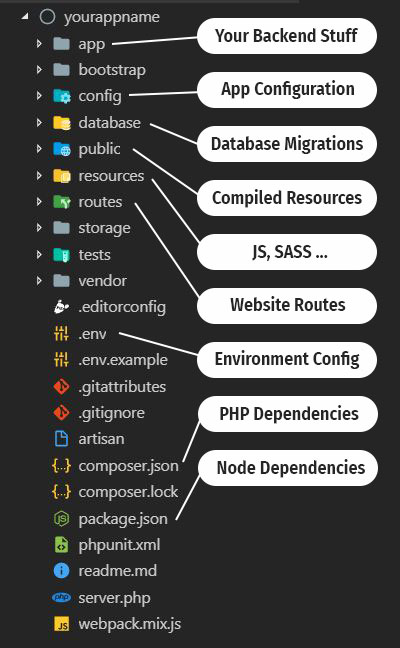


Figure 2: A New Laravel project directory structure (Hendricks, 2019).

## **Model-view-controller (MVC)**

Stands for "**Model-View-Controller**." MVC is an application design model comprised of three interconnected parts. They include the model (data), the view (user interface), and the controller (processes that handle input) (Raphael, 2014).

The MVC architecture pattern let the developer write a code that can be divided on the basis of the following three things:

### **Model**

A Model is the mode by which the developer can manipulate the data. It consists of a layer residing between the data and the application. The data itself can be stored in various types of database systems such as MySQL or even simple XML or Excel files (Smith, 2014).

### **Views**

Views are the visual representation of our web application (presentation layer), they are responsible for presenting the data that the Controller received from the Model (business logic). They can be easily built using the Blade template language that comes with Laravel or simply using plain PHP code. When Laravel renders these Views it examines first their file extension, and depending on it being either **“.blade.php”** or simply **“.php”**, determines if Laravel treats our View as a Blade template or not (Hendricks, 2019).

### **Control**

The primary function of a Controller is to handle requests and pass data from the Model to the Views. Thus, a Controller can be considered as the link between our Model and Views (Smith, 2014). The developer has the option to write her/his business logic either in Routers or Controllers. Routers can be useful when dealing with a small web application, or in rapidly defining static pages. Writing Controllers for every single page of the web application is thus not necessary (David, 2014).

### **Database**

**Eloquent ORM**

The Eloquent Object Relational Mapper (ORM), provided with Laravel includes a simple PHP ActiveRecord implementation which lets the developer issue database queries with a PHP syntax where instead of writing SQL code, methods are simply chained. Every table in the database possess a corresponding Model through which the developer interacts with said table (Nikko, 2012). The Laravel Schema class provides a database agnostic (i.e. can function with a multitude of DBMS) way of managing all database related work such as creating or deleting tables and adding fields to an existing table. It works with a multitude of databases systems supported by Laravel and MySQL being the default one. The Schema class has the same API across all of these database systems (Smith, 2014).

**Managing the database with Migrations**

Migrations can be considered as a form of version control for our database. They allow us to change the database schema and describe and record all those specific changes in a migration file. Each Migration is usually associated with a Schema Builder to effortlessly manage our application's database. A migration can also be reverted or “rolled back” using the same said file.

Using our terminal, we can issue the following commands to create or drop tables in our database:

**Table 1. A collection of commands related to migrations (Hardik, 2013).**

|  |  |
| --- | --- |
| **Command** | **Description** |
| $ php artisan migrate:install | Creates the migration repository |
| $ php artisan migrate:make | Creates a new migration file |
| $ php artisan migrate:refresh | Resets and reruns all the migrations |
| $ php artisan migrate:reset | Rollback all the database migrations |
| $ php artisan migrate:rollback | Rollback the last database migration |

## **Composer**

Another feature that makes Laravel stand out from the other frameworks is that it is Composer ready. In fact, Laravel is itself a mixture of different Composer components, this adds a much-needed interoperability to the framework. Composer is a dependency management tool for PHP. Essentially, Composer’s main role in the Laravel framework is that it manages the dependency of our project’s dependencies. For example, if one of the libraries we are using in our project is dependent on three other libraries and that there is a need to upgrade all those libraries, then there is no necessity to manually find and update any files. It is possible to update all four libraries via a single command through the command-line, which is, “$ composer update” (Smith, 2013).

**Advantages of Laravel Framework**

According to Shukla (2019), Laravel is the strongest contended in the PHP ecosystem simply because it has the following advantages.

1. **Template Tool**

Laravel has a pre-installed powerful and lightweight template engine, helping developers in making some extraordinary layouts with intensive content seeding. [Laravel template offers](https://laravel.com/docs/5.8/frontend) various solid widgets with robust CSS and JS coding. Its inbuilt templates are there to accumulate plain PHP and cached it to give the great performance. All the templates are designed with different sections along with simple layouts which make it easier to develop a simple yet functional layout.

1. **Authentication and Authorization**

Some of the best Laravel development companies assure secure authentication and authorization. Any web application owner needs to be assured of having authentic and authorized users accessing the secured resources. Implementation of authorization is quite simple as almost everything is configured out-of-the-box. Laravel also facilitates a basic method that sorts out the authorization logic and control access which are well-organized and very easy to manage.

1. **Inbuilt Libraries**

Laravel is the framework that facilitates you at best by being the only provider having dynamic pre-installed libraries. These libraries include Object Oriented libraries that cannot be found in other PHP frameworks. Besides offering hassle-free implementation, it also contains various features like watchword reset, checking dynamic clients, CSRF (Cross-site Request Forgery) insurance, and encryption. Additionally, Laravel has a helper functionality, helping you encourage and achieve various basic undertakings i.e. firing events, generating views, sending HTTP responses to the server etc. This is a significant feature that is found exclusively in Laravel.

1. **URL Generations**

Laravel also assist in generating URLs which becomes very helpful for building links in your templates.  All Laravel corridors are effectively laden by the framework which is delineated in the *app/Http/routes.php* file. When a user clicks or types a link, they want to see the desired content, such as an article, a product description, etc. which is not possible without the assistance of URL routing. As the web application will never understand what the user wants to see and may instead display a blank page or an error page.  Laravel framework provides a very plain route description strategy by simply accepting a URI and a Closure.

1. **Integration with Mail Services**

Mail service integration is another benefit that Laravel provides. It is used to send notifications to users to notify them about the various events that take place. It has become inconceivable nowadays, to have a modern online application that lacks simple email notification to new customers portraying their promising web registration. It also provides drivers for Mailgun, SMTP, Mandrill, SparkPost, PHP’s “mail” function and Amazon SES which allows an application to get started speedily.

1. **MVC Architecture Support**

While you google what Laravel is, Wikipedia’s definition states that it follows a Model – View – Controller architecture. And this is the factor which makes Laravel the best framework to use for your web application development. It enhances the performance, ensures clarity and allows for better documentation. It makes it easier to change the template and the underlying codes. In short, its built-in functionalities ease the overall process.

1. **Technical Vulnerabilities Fixture**

Laravel is a secure platform that takes care of the security within its framework. It safeguards web application against the most serious security risks such as; SQL injection, cross-site request forgery, cross-site scripting, etc. It can inform you first-hand if the codebase is guarded, and that many people have vetted the code. Therefore, any [Laravel development company](https://www.arpatech.com/laravel-development) is confident and will be sure to assure you of the framework’s security.

1. **Artisan Tool for Command Line**

Generally, a developer needs to interact with the Laravel framework using a command line that develops and manages the Laravel project environment. Laravel provides an integrated command – line tool called Artisan. This tool helps to create skeleton code and database architecture as well as their migrations. Database management becomes easier as a result. It enables the generation of basic MVC files and the formation of custom commands. Artisan tool allows us to perform almost all of those repetitive and tedious programming tasks that most developers avoid manually. Moreover, we can also conduct Unit tests for our application.

1. **Fine Unit Testing**

In Laravel’s unit testing, every module of your web application is tested before the site goes live so that no part of your website remains broken. Fine unit testing ensures that there are no bugs or exceptions in your web application which makes a high-performance and bug-free and ultimately hassle- free application for your end users.  It is another exception of Laravel framework.

1. **Multi-Lingual App Creator**

Apart from all the other benefits that Laravel’s development services provide, creation of multi-lingual apps is another one. It is indeed the right option for businesses looking to expand their reach across different countries with different languages. Laravel framework helps create your web application easily and quickly for different languages.

# Conclusion

After examining different PHP frameworks and looking at the various features of Laravel’s ability in handling MVC architecture pattern, I came up with the ideal choice for a PHP MVC framework, which is Laravel. At first, learning a new framework might seem an overwhelming task, but it’s not the case with Laravel, thanks to its clear and concise documentation, and its developers that make a lively active community.

**Recommendations**

1. Web app developers should employ the use of Laravel in developing their projects in other to make the project secure and robust.
2. Organizations or institutions should prioritise Laravel in when building their websites.

**REFERENCES**

Christensson, P. (2018). *MVC Definition*. Retrieved 2021, February 6, from <https://techterms.com>

David, C. (2014). *Bootstrap site blueprints,* Packt Publishing Limited, Birmingham.

Hardik D. (2013). *Learning Laravel 4 application development*. Packet Publishing Limited, Birmingham.

Hendricks, R. (2019). *Architecture of Laravel Applications*. Retrieved 2021, February 3, from <http://laravelbook.com/laravel-architecture>

James, E. (2019). *Framework Usage Statistics in the top 1 million website*, Retrieved 2021, February 6, from <https://morioh.com/p/bd8e04f90c30>

Nikko, B. (2012). *Build Web Apps from Scratch with Laravel*. Retrieved 2021, February 3, from <https://code.tutsplus.com/tutorials/build-web-apps-from-scratch-with-laravel-the-eloquent-orm--net-25631#:~:text=An%20ORM%20is%20an%20object,Laravel's%20built%2Din%20ORM%20implementation>

Raphael, S. (2014). *Getting Started with Laravel 4*. Packt Publishing Limited, Birmingham.

Shukla, S. (2019). *Key Benefits of using Laravel Framework.* Retrieved 2021, January 6, from <https://www.netsolutions.com/insights/laravel-framework-benefits/>

Smith, J. (2014). *Introduction to Laravel*. Retrieved 2021, January 6, from  [http://laravel.com/docs/introduction](%20http://laravel.com/docs/introduction%20on%206th%20January,%202021.%20)