A comparative study of laravel and symfony PHP frameworks

Majida Laaziri[1], Khaoula Benmoussa[2], Samira Khoulji [3], Kerkeb Mohamed Larbi[4], Abir El Yamami[5]
[1,2,3]Information System Engineering Resarch Group, National School of Applied Sciences,
Abdelmalek Essaadi University, Morocco
[4]Information System Engineering Resarch Group, Faculty of Sciences,
Abdelmalek Essaadi University, Morocco
[5]Laboratory Signals, Distributed Systems and Artificial Intelligence, ENSET Mohammedia,
Hassan II University, Morocco

Article Info	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 paper focuses on the elaboration of a comparative study between Laravel, symfony framworks, which are the most popular PHP frameworks. Thus, it provides an effective comparison model that merges seven dimensions: Features, Multilingual, System requirements, Technical architecture, Code Organization, Continuous Integration (CI) and finally Documentation and learning curve dimension. Results show that our model can be beneficial for IT project developers to select the suitable PHP Framework.

Corresponding Author:

Laaziri Majida,
Information System Engineering Resarch Group,
National School of Applied Sciences, Abdelmalek Essaâdi University,
Mhannech II, B.P 2121 Tetouan, Morocco.
Email: majida.laaziri@gmail.com

1. 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, ~~kly develop large~~ applications [1], by promoting code reuse, testing and changes to an application. There are many Frameworks that are popular and widely used for development written in different languages that are built around the same structuring facilitates the learning and understanding of a Frameworks, among them the PHP frameworks, including Laravel and symfony which are the most used by the developers. Each of these two frameworks has advantages and provides a specific implementation that should be taken into account to make its choice [2].

The author [2] claims that the Symfony framework is the best for complex enterprise-level applications, and to quickly and efficiently enrich the institution's information system. This is why he chose Symfony for the development of his project; which refers to an application to simplify the procedure of archiving the conference proceedings of researchers. Thus, he confirms that Symfony imposes structuring on its development; and it is adapted to an environment where turnover is important. The authors [3] in their

study of PHP frameworks declare that the Laravel framework makes it possible to develop the PHP code in an elegant and simple way, avoiding the "spaghetti code." And that the Symfony frameworks is a complete framework designed to optimize the PHP code, development of Web applications based on the Model View Controller pattern (MVC. The authors [4]declare in their analysis of PHP frameworks dedicated to the development of web information systems, that among the reasons for choosing the Symfony framework. It offers fast web application development and management, which makes it easy for web programmers to perform daily tasks, and supports multiple databases (MySQL, PostgreSQL, SQLite, etc.). Symfony is equipped with Doctrine, ORM, and the Symfony router has many options for configuring routes and annotations, YAML, XML and PHP [5].The authors [6] insist in their evaluation of performance of PHP frameworks, that Laravel is a framework that allows, on the one hand to work efficiently and is distinguished by the speed of its learning curve, it's very precise documentation and the support of the community as well. On the other hand, Laravel is better for large-scale web projects requiring faster delivery with fewer resources.

From the analysis of these studies, it is found that there is no precise method to effectively facilitate the choice between the two PHP frameworks: Symfony and Laravel and few authors who have explored this topic. For this reason a new complete model is proposed in this article containing a set of criteria to facilitate the choice between these two frameworks.

2.   LITERATURE REVIEW

The term "framework" in the case of Web frameworks, is a set of libraries and tools that can improve the design of Web applications, bringing additional features while adding rigor in their development. A framework also allows automation of tasks because it integrates a number of routines implemented natively. Using a framework is therefore guaranteed to have a coherent architecture, where the rigor of development is paramount. It is also a reflection on the future, since a light, optimized and consistent code is much easier to maintain and improve than the too famous "code soup" [7]–[9].

Not all frameworks meet the same needs, and several frameworks can be used together in some situations. We find application frameworks, for example web application frameworks (like Django, Ruby on Rails, Symfony or Catalyst) or desktop application development frameworks (Cocoa on Mac, Qt on cross platform ...). There are also frameworks specifically addressing certain development issues, such as logging frameworks (eg Log4J in Java), persistence frameworks and ORM frameworks (eg Hibbernate in Java, SQLAlchemy in Python, Propel in PHP...) as shown in Figure 1, web content presentation frameworks (Bootstrap, Foundation ...) [8], [9].
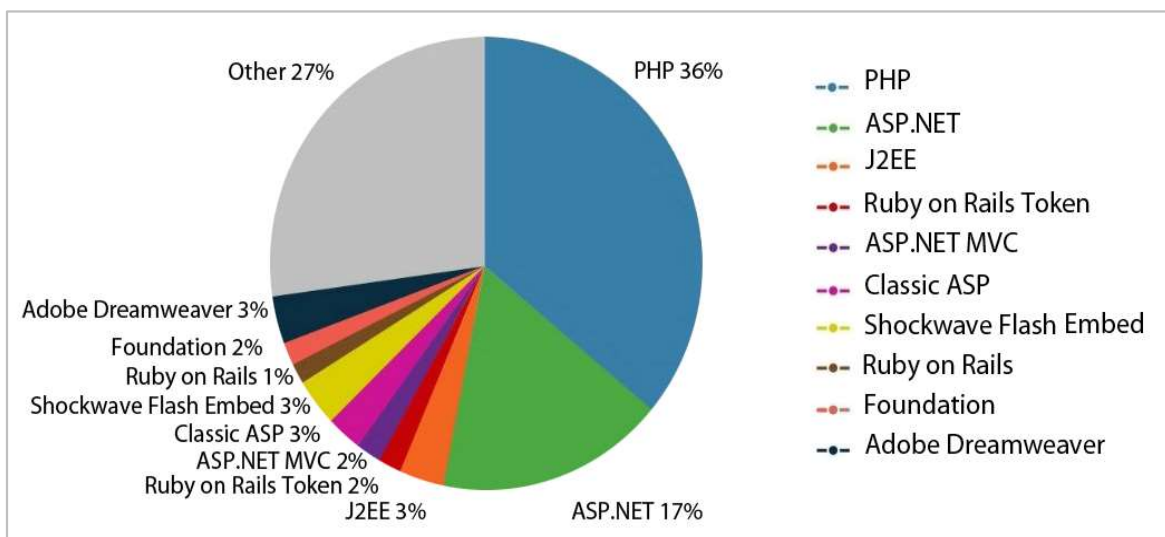

Figure 1. Framework Usage Statistics in the top 1 million website [10]

PHP frameworks streamline the development of web applications written in PHP by providing a basic structure for building web applications. In other words, PHP frameworks help promote Rapid Application Development (RAD), saving time, helping to create more stable applications and reducing the

amount of repetitive coding for developers [7]. Frameworks can also help beginners create more stable applications by ensuring good interaction and coding of the database on the presentation layer. This allows to spend more time creating the actual web application instead of spending time writing repetitive code. The general idea behind the functioning of a PHP framework is called the Model View Controller (MVC). MVC is an architectural programming model that isolates business logic from the user interface, making it possible to modify it separately (also called separation of concerns) [11].

Each type of PHP framework has its own advantages, several factors to look for in a PHP framework include: ease of use, fast development / performance, popularity among other developers, powerful features and support / forums. All frameworks are slightly different and have varying strengths and weaknesses[12], for example Zend Framework[13], created in March 2006 by Zend Technologies, is a free PHP framework created by Zend developers. full of features, and has an extensive support system since it has been around for so long. On the contrary, CakePHP [14] is another PHP framework that is younger than Zend Framework, its initial release was released in April 2005, CakePHP was just a modified project on Ruby on Rails and has a bit less of a system support (although support for this framework is growing rapidly), but is much more user-friendly and easy to use. Symfony [15] is one of the great frameworks, created in 2005 by the French Fabien Potencier, is an internationally recognized PHP framework that is quite complex but interesting for big projects. It should be noted that it is supported by a French team and therefore widely used in France; Laravel [16] is a framework created by Taylor Otwell in 2011, is based on Symfony 2, it quickly became one of the most popular PHP frameworks for web developers. Flexible, scalable and easy to use. Code Igniter [17]created in 2006 by the company EllisLab, is an ultra lightweight framework (a few hours are enough to learn how to use it). He suffers from the defects of his qualities, that is to say that he can be a little too simple on certain projects. Yii, [18] created by Qiang Xue in 2008, is a fast and extensible framwork, which helps professionals to work faster when developing PHP applications. Phalcon [19] was created by Andres Gutierrez in 2012, it is the fastest framework Developed in C language and distributed as an extension, this framework can not be used on standard shared hosting. According to many comparative studies, Phalcon displays a very great performance and it is, so far, the fastest, thanks to the template engine developed in C language. There are of course many other widely used. Let's mention Kohana [20], Solar [21], PHOCOA [22],etc.

Distinguishing Laravel framworks, symfony, which are actively maintained, popular with the developer community, promote modern web development practices and build web sites and web applications of all sizes and complexities. Choosing the right framework between different framwork, however, can sometimes be difficult. In order to make the selection process a little easier, we have proposed a model of analysis and comparison between different framworks, in our case we will choose laravel and symfony which are the most used and popular.

3.   PROPOSED MODEL

To make the selection process between PHP frameworks (in our case: between the two PHP frameworks Symfony and Laravel) easier for developers of information systems, we propose a new holistic model that groups together a set of dimensions: Features, Multilingual , System requirements, Technical architecture, Organization code, Continuous Integration (CI), Documentation and learning curve. Each dimension is characterized by a set of criteria mentioned in Table 1. This model makes it possible to compare the two PHP frameworks in two steps:

Before starting the comparison between Symfony and Laravel we start with what they have in common since they come from the same PHP programming language, we analyze the characteristics of each of them. Then we make a detailed study of the two frameworks through the following dimensions:

a. Technical architecture: each framework offers a different technical structure to the other, including the organization of the directory and the files, the installation, the bundles, the library, etc.
b. Code Organization: refers to the structuring of the code of each PHP framework.
c. Continuous Integration (CI): Whenever changes are made to the code, a revision is required, along with some tests and reconfigurations. CI (Continuous Integration) is a practice that automates such tasks aimed at early mistakes. There are different servers and continuous integration tools in the PHP environment: Travis Ci, CI Style, Drone, Codeship, Circle CI, Jenkins
d. Documentation and learning curve: concerns learning materials and documentation, tutorials and support videos of each PHP framework.

Table 1. Proposed Model

Features	Scafollding Full text search
Multilingual	Multilingual content
System requirements	Operating system
	Programming language
	Tag
	Multi-user system
	Autofocus
	Pingback
	Extension/Plug-in
	Image processing engine
	WYSIWYG-Editor
	Multiple projects
	External pages
	User statistics
	Revision control
	PSR-0 compliant
	Machine Code Generation
	Programming language
	Installation
	The core
	Access to the base
	Service container
	Model engine
Technical architecture	Forms and validators
	Cache and performance
	Debugging and development tools
	Administration Panel
	Search engine
	Event Dispatcher and Middleware
	REST API
	Sentinel
	patterns
	Compatibility of the REST API
	Scaffolding
	MVC
	HTML modeling
Code Organization	Installation via the composer
	ORM
	Multiply drivers support DB, storage
	Full text search
	CI Support, AQ
	Travis Ci
	CI Style
	Drone
Continuous Integration (CI)	Codeship
	Circle CI
	Jenkins
	Learning curve
	Complexity of installation
Documentation and learning curve	Events / Community
	Documentation
	certification

4. DISCUSSION OF RESULTS

We present below an instantiation of our proposed model for the comparison of Laravel and Symfony frameworks, this study is based on the results obtained in [23], [24]: Since the two frameworks come from the same context (i.e. PHP), there are many crucial features that are present in both of them. After extensive research, we found that Symfony and Laravel both support Scaffolding and full-text search. Cross-platform functionality and multilingual content acceptance are also common in both cases [25],[26]. We present trough sections A, B, C (Table 2, Table 3, Table 4) these similarities.

a. Features

Table 2 presents the characteristics of each framework according to features dimension

Table 2. Comparison According to Features Dimension

r	Symfony	Laravel
Features		
Scafollding	Yes	Yes
Full text search	Yes	Yes

A comparative study of laravel and symfony PHP frameworks (Majida Laaziri)

b. Multilingual
   Table 3 presents the characteristics of each framework according to multilingual dimension

Table 3. Comparison According to Multilingual Dimension

Multilingual	Symfony	Laravel
Multilingual content	Yes	Yes

c. System Requirements
   Table 4 presents the characteristics of each framework according to system requirements dimension

Table 4. Comparison According to System Requirements Dimension

System requirements	Symfony	Laravel
Operating system	Cross-platform	Cross-platform
Programming language	PHP	PHP
Tag	MVC,Dependency,Injection, Namespaces	MVC,Dependency,Injection, Namespaces
Multi-user system	Yes	Yes
Autofocus	Yes	Yes
Pingback	Yes	Yes
Extension/Plug-in	Yes	Yes
Image processing engin	Yes	Yes
WYSIWYG-Editor	No	No
Multiple projects	Yes	Yes
External pages	Yes	Yes
User statistics	Yes	Yes
Revision control	Yes	Yes
PSR-0 compliant	Yes	Yes
MachineCode Generation	Yes	Yes

d. Technical Architecture
   Table 5 presents the characteristics of each framework according to technical architecture dimension

Table 5. Comparison According to Technical Architecture Dimension

Technical architecture	Symfony	Laravel
Programming language	PHP	PHP
Installation	Composer	Composer create-project or via Laravel Installer
The core	The Symfony Components	The Symfony Components
Access to the base	Doctrine	Eloquent
Service container	injections	App :: make () or resolve () functions
Model engine	Twig	Blade
Forms and validators	model	form or via the manual validation of a request
Cache and performance	yes	yes
Debugging and development tools	Advanced	Moderate
Administration Panel	Advanced	Moderate
Search engine	FOSElastica and KnpBundles	laravel-elasticsearch and laravel-solarium
Event Dispatcher and Middleware	yes	yes
REST API	FOSRestBundle and JMSSerializerBundle packages	routes/api.phpfile.
Sentinel	sentry-symfony	laravel-raven

   Results in Table 5 show that both frameworks use the php language, and installed via Composer. For Symfony, the role of the composer is more crucial. The idea of component management is best achieved by using the PHP Composer dependency manager. After installation, Symfony 2 also provides a demo application to get started. Laravel is also easy to install using Composer create-project or via Laravel Installer.

Laravel and Symfony share the core. Both frameworks are built on top of the library set called "The Symfony Components". Laravel has added its own components in addition to what Symfony provides, fixing issues and adding features that may be missing.

Both frameworks access the database in a different way. Symfony uses Doctrine, while Laravel uses Eloquent.The symfony service container is a way to access one service from another by using so-called injections. For Laravel there is another way to access the services, it is to use the App :: make () or resolve () functions.

Laravel comes with a template engine called Blade and Symfony bundled with Twig. Both template engines implement two main features : a). Model inheritance, b). Blocks or sections

Both features allow you to define a basic template with replaceable sections and child templates that populate the values in those sections. Form validation is a big difference between Symfony and Laravel. In Laravel, the validation can be done in form or via the manual validation of a request, whereas in Symfony the developer can validate only one model.

Symfony and Laravel cache their views, but only Symfony caches the default source code.

Debugging is a necessity for every application. Good IDE support, as well as debugging and profiling tools, help the developer find problems in their code much faster. When it comes to IDE, Symfony has better support. The support for Laravel is very minimalist, but it has a library called laravel-ide-helper that fills most holes.

Symfony has a very advanced panel showing most of the problems, as well as profiling details. Laravel has only one panel that can display exceptions and do basic profiling. Symfony contains a standardized and very advanced administration panel, Laravel has some packages that can be used as an administration panel.

The use of elasticsearch or Solr has become a must on modern websites. Symfony and Laravel both support using third-party packages: FOSElastica and KnpBundles for Symfony, laravel-elasticsearch and laravel-solarium for Laravel.

- **Symfony and Event Dispatcher**

To convert a request into a response, Symfony uses EventDispatcher. This triggers different life cycle events and special event listeners to handle these events. At first, it distributes the kernel.request event that includes the request information. The default primary listener for this event is RouterListener, which calls the router component to find an appropriate routing rule for the current request. After that, other events are executed step by step. Typical event listeners are a security check, a CSRF token check, and a logging process. If you want to add certain features to the request lifecycle, you must create a custom EventListener to subscribe it to the necessary event.

- **Laravel and Middleware**

Laravel uses a different solution: middleware. The developer application has some layers and a request passes through these layers on the controller path and vice versa. So, if the developer wants to extend the logic of his application and adds some functionality in the query lifecycle, he has to add an extra layer to his middleware list, and Laravel will execute it as shown in Figure 2.
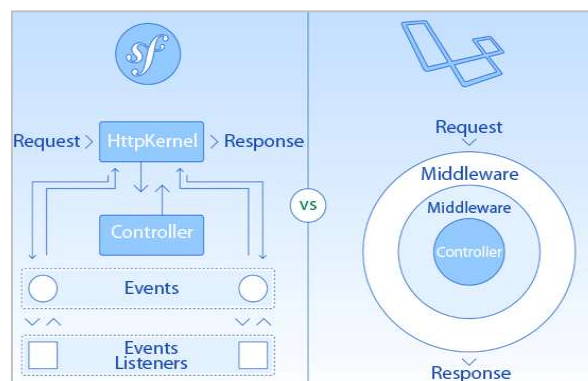

Figure 2. Event dispatcher and middleware

Symfony does not have an easy out-the-box solution for creating fast REST APIs, but it has large third-party FOSRestBundle and JMSSerializerBundle packages. In Laravel, the developer must first define routes. It can do this in the routing rules api section to disable some default middleware components and activate others. The api section is located in the routes/api.phpfile.

Some websites may need to use more space than a host can provide. The typical solution in such a case uses either Amazon S3 or another cloud system. Symfony manages these external drives using a wafer-based third-party package (13 adapters), while Laravel manages them using flysystem-based internal libraries (20 adapters).

Debugging is difficult if the developer does not know what caused a problem (or when it happened) - that's why the Sentinel is so popular. Symfony accesses the sentry using sentry-symfony, while Laravel uses laravel-raven.

e.   Code Organization
Table 6 presents the characteristics of each framework according to code organization dimension

Table 6. Comparison According to Code Organization Dimension

Parameter	Symfony	Laravel
patterns	Factory, Composite, QueryBuilder, Flyweight, Observer, Dependency Injection, Data Mapper	Builder, Factory, Depot, Strategy, Supplier, Facade, ActiveRecord, Dependency Injection
Compatibility of the REST API	Yes (with FOSRestBundle)	Yes
Scaffolding	Yes	Yes
MVC	Yes	Yes
HTML modeling	Brindille	Lame
Installation via the composer	Yes	Yes
PHP standards recommendations	Yes	Yes
ORM	Yes	Yes
Multiply drivers support DB, storage	Yes	Yes
Full text search	Yes (ElasticSearch)	Yes (ElasticSearch)
CI Support, AQ	No	PHPUnit

As far as learning materials and documentation are concerned, Laravel's tutorials and support videos (Laracasts) cover everything from A to Z, and the documents in their Laravel Github documents are numerous. The Symfony documentation is of course also available to explain its components, bundles, service containers and more.

Further, Laravel is better for large-scale Web projects that require faster delivery with fewer resources at the same time as shown in Figure 4, Symfony is better for complex applications at the enterprise level, and also requires developers with superior skills as shown in Figure 3. To summarize, the main advantages of the Symfony web framework are:
• Several options for extensibility
• Faster than most other PHP frameworks
• Adaptability and optimal performance (request/response centered on HTTP)
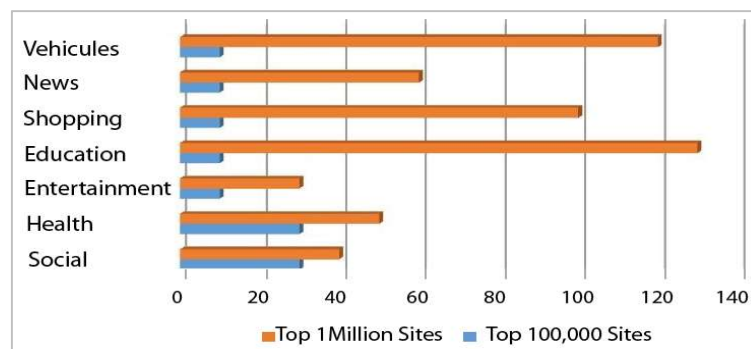• Reusable components that reduce time and cost

Figure 3. Symfony PHP framework popular website verticals [10]

And the main advantages of the Laravel web framework are:
▪ Advanced query builder mechanism.

- Painless data migration and global management
- Automatic loading installation (no need for manual maintenance)
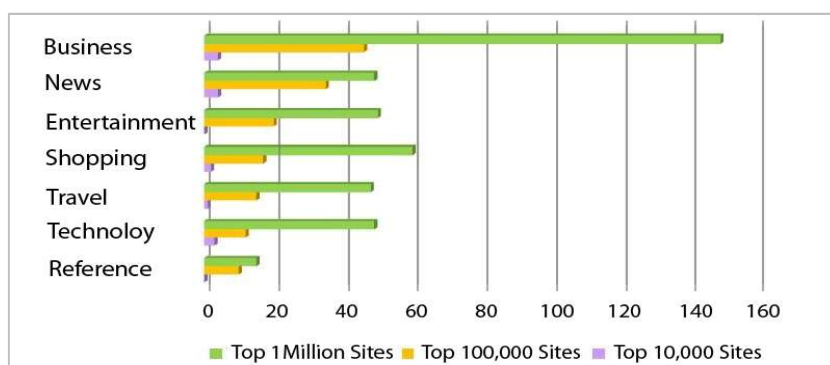- Simplified authentication

Figure 4. Laravel PHP framework popular website verticals [10]

The two graphs in Figure 3 and Figure 4 shows the number of websites belonging to the largest groups of sites (10,000, 100,000, and 1,000,000) in the specific vertical category.

5.   CONCLUSION

This paper provided an effective model for the comparison and analysis of the most used PHP frameworks: Symfony and Laravel. The results show that the two PHP-based frameworks are excellent viable options for most PHP-based projects, and provide a full-stack web application development environment for developers. Symfony may be more suitable for larger projects and is considered the most stable PHP framework supported by an extended community and LTS. Laravel, for its part, is the most popular framework for the development of the complete stack and has by far the flatter learning curve of all the frameworks. He comes with LTS and the support of the community too. Therefore, choosing a PHP framework should depend on resources and long-term development plans and to help the developer choose the best.

REFERENCES

[1] D. R. Lakshmi and S. S. Mallika, "A Review on Web Application Testing and its Current Research Directions," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 7, no. 4, p. 2132, 2017.

[2] B. Ninassi and S. Panay, "Les Frameworks De Développement Web : Comment Enrichir Rapidement Et Efficacement Le Système D ' Information De Votre Etablissement," pp. 1–8, 2011.

[3] F. Sierra, J. Acosta, and J. Ariza, "Estudio Y Análisis De Los Framework En Php Basados En El Modelo Vista Controlador Para El Desarrollo De Software Orientado A La Web," *Rev. Investig. y Desarro. en TIC*, vol. 4, no. 2, pp. 1–13, 2013.

[4] N. Prokofyeva and V. Boltunova, "Analysis and Practical Application of PHP Frameworks in Development of Web Information Systems," *Procedia Comput. Sci.*, vol. 104, no. December 2016, pp. 51–56, 2016.

[5] S. Subramaniam, S.-C. Haw, and P. K. Hoong, "Bridging XML and Relational Databases: An Effective Mapping Scheme based on Persistent Labeling," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 2, no. 2, pp. 239–246, 2011.

[6] R. Das and D. Prasad Saikia, "Comparison of Procedural PHP with Codeigniter and Laravel Framework," *Int. J. Curr. Trends Eng. Res. Sci. J. Impact Factor*, vol. 2, no. 6, pp. 42–48, 2016.

[7] L. Blanc, E. Gouleau, O. Mansour, T. Rivoallan, V. Lemaire, and X. Lacot, "Frameworks PHP pour l'entreprise," pp. 1–38.

[8] N. Richeton and C. Version, "Frameworks PHP," *Consultant*, pp. 1–78.

[9] V. Petrosyan, "A Guide to Popular PHP Frameworks for Beginners | SEJ," 2016. .

[10] "Web Technology Usage Trends Web and Internet Technology Usage Statistics." [Online]. Available: https://trends.builtwith.com/.

[11] R. F. Olanrewaju, T. Islam, and N. Ali, "An Empirical Study Of The Evolution Of PHP MVC Framework," in *Lecture Notes in Electrical Engineering*, vol. 315, pp. 399–410, 2015.

[12] V. Narayana, P. Premchand, and A. Govardhan, "Performance and Comparative Analysis of the Two Contrary Approaches for Detecting Near Duplicate Web Documents in Web Crawling," *International Journal of Electrical*

and Computer Engineering (IJECE), vol. 2, no. 6, pp. 819–830, 2012.

[13] "Home - Zend Framework." [Online]. Available: https://framework.zend.com/.

[14] "CakePHP - Build fast, grow solid | PHP Framework | Home." [Online]. Available: https://cakephp.org/.

[15] "Symfony, High Performance PHP Framework for Web Development." [Online]. Available: https://symfony.com/.

[16] "Laravel - The PHP Framework For Web Artisans." [Online]. Available: https://laravel.com/.

[17] "CodeIgniter Web Framework." [Online]. Available: https://codeigniter.com/.

[18] "YiiFramework." [Online]. Available: https://www.yiiframework.com/.

[19] "Un framework PHP ultra-performant - Phalcon Framework." [Online]. Available: https://phalconphp.com/fr/.

[20] "Kohana: The Swift PHP Framework." [Online]. Available: https://kohanaframework.org/.

[21] "Solar Framework for PHP 5." [Online]. Available: http://solarphp.com/.

[22] "The phocoa Open Source Project on Open Hub." [Online]. Available: https://www.openhub.net/p/phocoa.

[23] "Laravel vs Symfony - Clash of the Frameworks." [Online]. Available: https://www.merixstudio.com/blog/laravel-vs-symfony-clash-frameworks/.

[24] "Difference Between Symfony vs Laravel | Choose between Symfony vs Laravel - Omniceps." [Online]. Available: http://www.omniceps.com/symphony-vs-laravel-differences-better/.

[25] "Symfony or Laravel: Which PHP Framework Should You Choose - DZone Web Dev." [Online]. Available: https://dzone.com/articles/symfony-or-laravel-which-php-framework-to-choose.

[26] "Symfony   vs   Laravel :   Which   PHP   framework   to   choose?"   [Online].   Available: https://www.valuecoders.com/blog/technology-and-apps/symfony-vs-laravel-php-framework-choose/.