**PHP FRAMEWORKS**

Amanda Barry

Interactive Media Department  
University of Miami

Miami FL USA

[apbarry30@miami.edu](mailto:apbarry30@miami.edu)

**ABSTRACT**

PHP (Hypertext Preprocessor) scripting language is widely used by developers to create web applications. Frameworks of PHP assist in creating these applications with scripts to speed-up processes and save the developer a lot of time. To choose the best framework, a developer needs to consider the purpose of their application and what processes are necessary to make it work. PHP frameworks allow for programmers to focus on key functionalities specific to developing their application and progress faster.

**KEYWORDS**

php, frameworks, laravel, symfony, web, applications, language

**1 Background**

PHP (Hypertext Preprocessor) is an open-source scripting language that can easily be integrated into HTML. PHP is widely used and functions with all major operating systems. PHP’s main focus is server-side scripting, forming a server-side cache for your dynamic content, but is not limited to that as it also is used in command line scripting and writing desktop applications. There are many PHP frameworks that can be utilized for assisting the development of web apps and speed up the development process. There are pros and cons to different frameworks depending the goal of your application. The framework that should be used is based on the framework’s ability to facilitate the apps functioning and development. Laravel and Symfony are two of the most popular frameworks among PHP developers.

* 1. **USING LARAVEL**

Laravel has gained popularity because of its focus on security and ability to handle complex web applications securely at a faster pace than other frameworks. Laravel targets B2B or enterprise websites that evolve with changing web trends. Laravel is built to ease the process of common tasks related to high-security needs such as routing, sessions, caching, and authentication. The security is up to date with expectations for a modern web application. Laravel is easy to install and works well for developing applications with complex back-end necessities. In addition to the security feature of Lavavel, other features include seamless data migration, MVC architecture support, routing, view template engine, and authentication, among many others. All of these features aid in customizing your application even further. The syntax is easy to use and the functionality core can be extended. Laravel can be easily integrated with third-party libraries and unit testing is supported but it cannot run more complex routing scenarios.

* 1. **USING SYMFONY**

Symfony is a mature and reliable platform that is one of the oldest of the frameworks. Symfony is an extensive PHP MVC framework used for developing large-scale enterprise projects. Symfony targets advanced developers and the mechanisms can be harder to use. Symfony is easy to install but because it has so many functions, it can be slower compared to other frameworks. A key feature is its reusable PHP components. Symfony has a stable database-engine that complies with most web best practices and design patterns, allowing for integration with other vendor libraries. It is the only framework known to follow PHP and web standards to a tee. The MVC architecture allows for a very flexible framework, enabling full control over configuration and other critical features of development of your application. Symfony offers easy unit testing through the Unit Independent Library. These tools for functional, behavioral and unit testing saves a significant amount of time and effort of the developer. There are also tools to resolve coding errors and security issues. Symfony allows for developers focus more on the core and advanced functionalities of the application rather than basic features.

* 1. **COMPARE AND CONTRAST**

Both the Laravel and Symfony frameworks provide object-relational mapping (ORM) but handles them in different ways. There both result in data manipulation becoming much simplier in either framework. There are some database-related differences between the two frameworks in terms of what they support. The databases Symfony supports include: Drizze, MySQL, Oracle, PostgreSQL, SAP Sybase SQL Anywhere, SQLit and SQLServer. In comparison, Laravel supports all of these except SAP Sybase SQL Anywhere, PostgreSQL and Drizzle. If you are going for breadth, Symfony is better. In regards to PHP as a templating engine, Laravel’s default engine provides benefits over Symfony as it allows for code reusability and a sub-framework called Lumen for building APIs.

**CONCLUSION**

The PHP frameworks, Laravel and Symfony have many similar features, and choosing one over the other is dependent on what the key functionalities of your application are and where you are experiencing problems. Both frameworks are successful in running a secure and complex web-application while providing users comfort by saving much time and effort in development.

**REFERENCES:**

1. Laravel. **“Homestead.”** Laravel, <https://laravel.com/docs/4.2/homestead>
2. Njenga. **“10 Popular PHP frameworks in 2019.”** Raygun, <https://raygun.com/blog/top-php-frameworks/>
3. The PHP Group. **“What Can PHP Do?”** PHP,<https://www.php.net/manual/en/intro-whatcando.php>
4. Froncza. “Laravel Vs. Symfony: PHP Framework Comparison”. Stackify, <https://stackify.com/laravel-vs-symfony-php-framework-comparison/>