COMPARISON BETWEEN YII FRAMEWORKS AND LARAVEL IN 3 DIFFERENT VERSION FOR VIEWING LARGE DATA OF SHIPYARD INDUSTRY IN INDONESIA

Ummi Khaira Latif[1], and Tien Fabrianti Kusumasari[2]

Department of Industrial Engineering Telkom University[1,2] Information System, Department of Industrial Engineering Telkom University[1,2] khaira.latif@gmail.com[1], tien.kusumasari@gmail.com[2]

Abstract—The objective of this paper is to compare two PHP frameworks with execution time, peak memory usage and throughput metrics. This performance comparison was done for Laravel and Yii2 frameworks using MySQL employee database [26] which freely downloaded for testing necessary. The database provided a large amount of data included million entry records. Using a large amount of record, we try to give some view to another developer for considering when choosing the PHP framework for their development. The result in this paper give Laravel has better performance in execution time and throughput measurement but not in memory usage. This result can be affected by many factors including the testing environment in when comparison process implementation.

Keywords—Laravel, Yii2, PHP Framework, Performance Comparison.

I. INTRODUCTION

PHP is a server-side scripting language designed specifically for web-based applications [1]. There are many advantages of the PHP language for example, performance, scalability, open source, portability, etc. According to w3techs.com PHP is the most commonly used scripting languages on the Internet with 82% coverage [2].

Instead of the advantages, PHP has some lack in the usage of some users. Very often in query database process, some users who are not familiar with the syntax of query languages such as SQL are not interesting on using complicated query language. The most alternative way to choose is using traditional textual query language which is easy to read visually to extract the information from the database [3] and using the direct manipulation mechanism [4]. The system could be vulnerable if it has often a connection to external resources like databases to handle large datasets [5]. Those mechanisms which are used inside the system becoming one major disadvantage in PHP.

Another problem such as script vulnerabilities can be occur. In other research [6] explain that PHP scripts are not inherently bad, the user could either inadvertently write a script that allows the vulnerability or maliciously create a script to take advantage of certain PHP functions.

For avoiding inconsistency programmable techniques done by the user, a more effective way to achieve quality software is to use frameworks [7]-[8]. Explanation of a framework was defined as a reusable, "semi-complete'' application that can be specialized to produce custom applications [9] was referred from [10]-[11]. In some of the software development, some reuse function also provided by the framework [12].

Nowadays many PHP frameworks available and easy to use for developing process. According to [13], seven frameworks are chosen to be the most popular in 2013. Those seven frameworks are CakePHP (version 2.5.3), CodeIgniter (version 2.2.0), Laravel (version 4.2), Symfony (version 2.5.3), Yii (version 1.1.15), Zend Framework (version 2.3.2) represented in Fig 1.
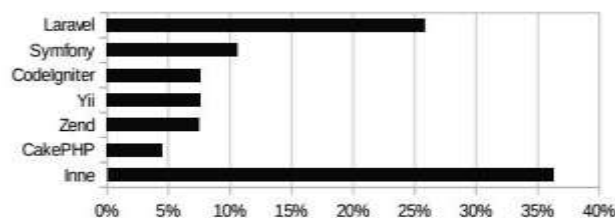

Fig. 1 The most popular frameworks in 2013 [13]

From those seven most popular framework, we will choose two frameworks as our focused comparison for this paper.

II. RELATED WORK

There have been many studies evaluating the comparison of some PHP framework. Some of those studies also focusing on PHP framework performance. For better of this study, understanding first other studies are important.

In 2016 June, Ripunjit and Dr. Lakshmi studied the comparison of procedural PHP with Code Igniter and Laravel framework. The objective of their study is to evaluate the performance of procedural PHP with CodeIgniter and Laravel framework. The decision of making the study is based on the fact that there is a lack of comparison tests between the popular PHP frameworks like Laravel, Codeigniter with plain PHP.

The resulting outcome of the experiment has been analyzed and interpreted. The study analyzes the performance based on the execution time also the memory usage of each framework [15].

In 2015, John Samra also studied about the performance comparison of PHP framework. With his thesis title "Comparing Performance of Plain PHP and Four of Its Popular Framework". The objective of his study is to evaluate the performance of four popular PHP frameworks Laravel, Symfony, CodeIgniter, and Phalcon together with the plain PHP. The reason for making this study was based on the fact that he thinks there is a lack of comparison tests between the most popular PHP frameworks. He also visiting the official websites of these frameworks, the first thing to notice is the slogans that have been made by the core teams. The majority of these slogans contain quality attributes like speed and high performance. Therefore, he put the performance of these frameworks to the test. And to do that, he did three experiments was conducted in which five functionally equivalent PHP applications are developed and used as targets. The experiments are conducted in two sessions. The first session deals with the execution time and the stack trace measurements while the second one is covering the measurement of the memory usage consumption [16].

III. TESTING ENVIRONMENT

We implemented our system on Windows 10. All the timing results in the paper generated on an Intel Core i5 and 12 GB of memory. We demonstrate the performance of our algorithm on the following scenario.

3.1.Database

We use employee dataset example from MySQL. MySQL provides large dataset for user testing necessary. We can found in the MySQL official website and download it freely from the GitHub repository [26]. For this performance testing, we choose two tables with different amount of data. The two tables are employee and salary table. The employee's table consists of 300.024 entry records. The second table has a bigger amount of records, it is consisting of 2.844.047 entry records. Those two tables were chosen by a number of records consideration. For getting the proper comparison, we need a big number of data to be executed by the system.

3.2.PHP Framework Consideration

Web application frameworks are available in almost web programming language including PHP, ASP.NET, C++, Perl, Ruby, Java, Python. Originally

started in 1994 to help streamline basic website tasks, PHP is a widely used, since becoming one of the most used website programming languages with nearly 20 million installations worldwide[17]. Based on PHP's popularity, there are so many Frameworks are available and easy used provided for development. In [19] cited by [18], shows that currently almost 40% of the top million sites on the internet was distributed by PHP framework as shown in Fig 2.
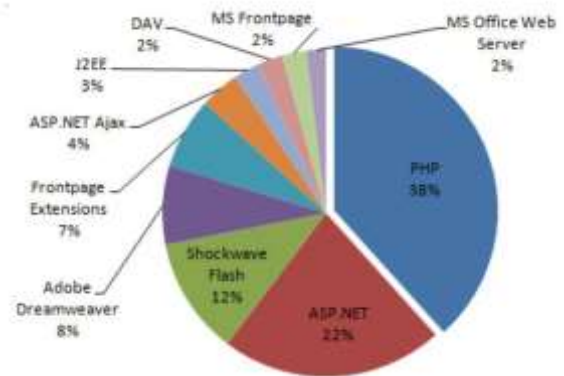

Fig. 2 Framework Distribution in Top Million Sites [19].

In selecting a framework for this research, we consider of the usage trend in recent years, popularity among the development communities, clearly of the documentation as the guidance of the development. As the time of writing this paper, the 5 popularity PHP frameworks listed in [13]. From those 5 top frameworks, we choose two frameworks they are Laravel and Yii2 as our concern of performance comparison in this paper.

3.3.Laravel

Laravel as the most popular PHP framework based on [13] was created by Taylor Otwell. The first released is Laravel version 1 in June 2011 [20]. In the time this paper was written, Laravel has released the newest version on August 30, 2017. Many improvements and capabilities were improved in each version of Laravel. In Laravel version 5 has some main features that explained in the [21], they are:

- Modularity: Laravel was built on top of over 20 different libraries and able to split into individual modules. It is also tightly integrated with Composer dependency manager these components can be updated with ease.
- Testability: It is easy to build from the ground for testing necessary. Laravel ships with several helpers that let you visit routes from the test, crawl the resulting HTML, ensure that methods are called on certain classes, and even impersonate authenticated users in order to make sure the right code is run at the right time

- Routing: Laravel gives you a lot of flexibility when you define the routes of your application.
- Configuration management: Laravel has a consistent approach to handle configuration settings, and different settings can be applied in different environments via the use of an .env file, containing settings unique for that environment.
- Query builder and ORM: Laravel ships with a fluent query builder. In addition, it provides with an Object Relational Mapper (ORM) and ActiveRecord implementation.
- Scheme builder, migration, and seeding: this feature allows you to define your database schema in PHP code and keep track of any changes with the help of database migration.
- Etc.

Laravel is a framework that is based on Model View Controller (MVC) paradigm. MVC understanding was explained in [22]. The following diagram was illustrated MVC interaction that applied in a typical web application.
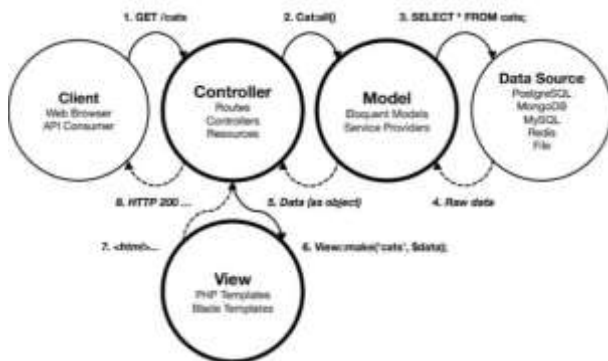

Fig. 3 MVC interaction in web application [21].

3.4. Yii

Yii is pronounced as Yee or [ji:], and is an acronym for "Yes It Is!". Yii is a free, open-source Web application development framework written in PHP5 that promotes clean, DRY design and encourages rapid development. It works to streamline your application development and helps to ensure an extremely efficient, extensible, and maintainable end product [23].

Yii also based on MVC paradigm and already explained in [22]. MVC aims to separate business logic from user interface considerations so that developers can more easily change each part without affecting the other.

3.5. Environment Implementation

In this paper, Laravel and Yii2 framework will be run inside Apache web server [24]. PHP version 5.6 and PHP version 7.1 was used for performance testing. This two version was chosen as one of the considerations for defining the better framework implementation environment factor. There are many developers review the performance difference between PHP version 5.6 and 7.1 in the part of speed, system support, and other performance. Therefore, the testing in this paper also implemented on that two version for showing the better performance and compatibility for executing large amount data records from the database.

In this paper, we installed Laravel version 5.2 and Yii version 2 as the framework version comparison. For measure the performance, we calculate the performance result shown by each debug bar. Yii version 2 already included debug toolbar in the installer packet, different with Laravel version 5.2. In the Laravel implementation, we included debug bar plugin developed by Barry vd. Heuvel [25].

In each debug bar, we take execution time and maximum memory usage or peak memory in each process. We then narrowed down the outcome number and make the comparison followed by the conclusion result of the process.

IV. RESULT

The result will be delivered depend on the number records execution and PHP version used in the testing process. Number record tested are employee table with 300.024 entry and salary table with 2.844.047 entry.

4.2. Execution Time Comparison in PHP 7

In this comparison, we will compare the result of execution time take in PHP version 7 with salary's table. We did six times execution continuously in each testing process. The lower number in the graph better results.
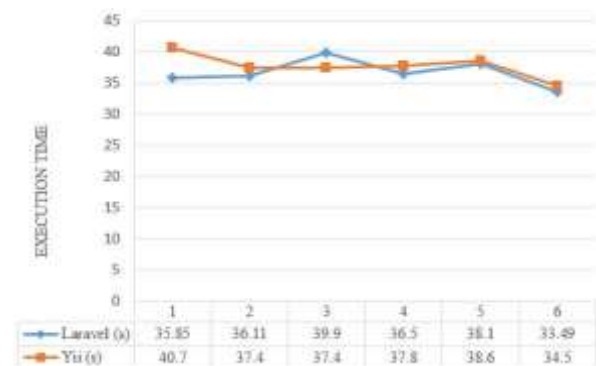

Fig. 4 Execution time take in PHP 7 using salary's table

Figure 4 shows the result of the execution time of salary's table. The lower number in the graph is better results. In this test, Laravel and Yii2 framework

execute over two million records. We can see that there is no much different time between those frameworks. In six times continuously execute the process, we get the average of those result shown in Fig. 5. The average of those execution time has about 1-second different. Laravel 1 second faster than Yii. For this testing, Laravel and Yii2 framework don't have much different performance in executing large record data.
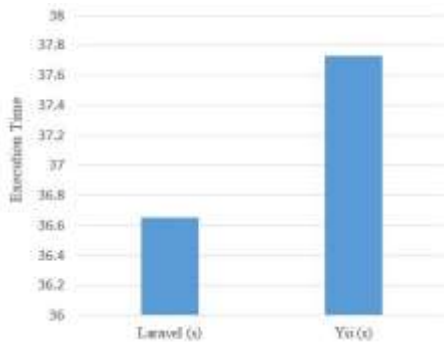

Fig. 5 Average comparison of execution time take in PHP 7 using salary's table

4.2. Peak Memory Consumption Comparison

Understand memory consumption needed in executing large data is needed when you want to choose the best framework for developing an application with this function.
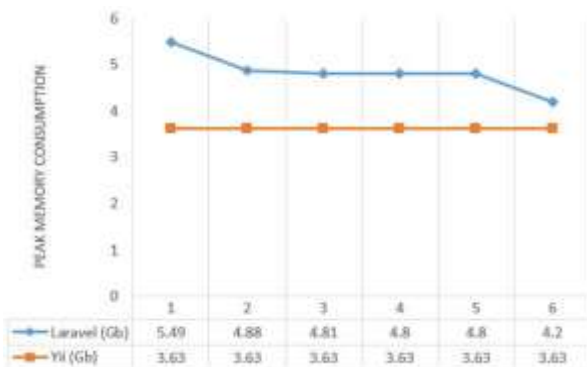

Fig. 6 Peak memory consumption in PHP 7 using salary table's

Memory consumption in Laravel and Yii2 testing result shows in Fig. 6. From those result, we can conclude that Yii2 has a stable peak memory consumption in each execution different with Laravel. The graph result shows that memory consumption in Laravel change in almost all the execution process.

In this measurement, we got the average of peak memory usage from those frameworks in Fig. 7. From those comparison result, we can say that in six continues execution process, Yii2 has lower peak memory usage than Laravel.
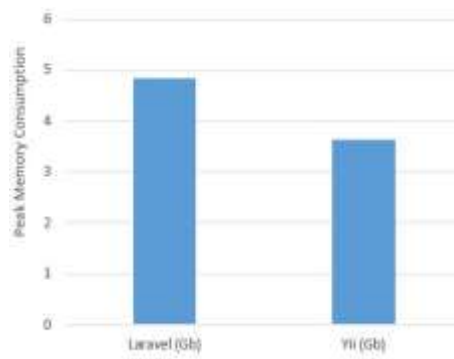

Fig. 7 Average Comparison of Peak memory consumption in PHP 7 using salary table's

4.3. Framework Implementation in PHP version 5 and PHP version 7

In this testing, we compare the execution time and peak memory usage of two frameworks running using PHP 5 and PHP 7. This testing was done using employee's table data around 300 thousand records. This records number is lower than salary's record number. PHP 5 is fails to execute salary's table because of too big data amount. We try to increase maximum memory usage allocation. But the maximum memory allocation in PHP 5 only about 7 Gb according to the author's testing environment. Therefore, in this testing process, we use lower number record.
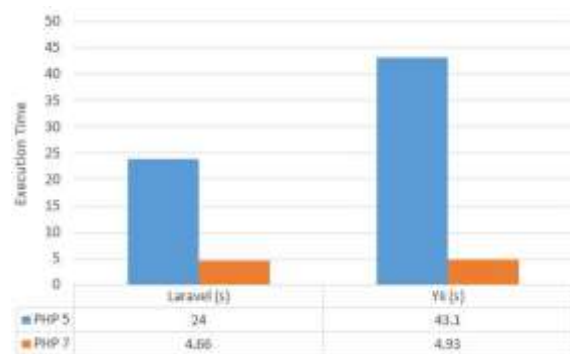

Fig. 8 Average Comparison of execution time between PHP 5 and PHP 7 using employee's table

Figure 8 shows the comparison results, the lower number has the better performance in this case. Based on that graph, we can see that implemented framework in the newest version PHP give much better performance in execution time.

For memory usage comparison, we also get better performance in PHP version 7(see Fig. 9). The peak memory usage was much decreased from PHP version 5 to PHP version 7. Many of the developers also give the same result of this similar comparison [14].
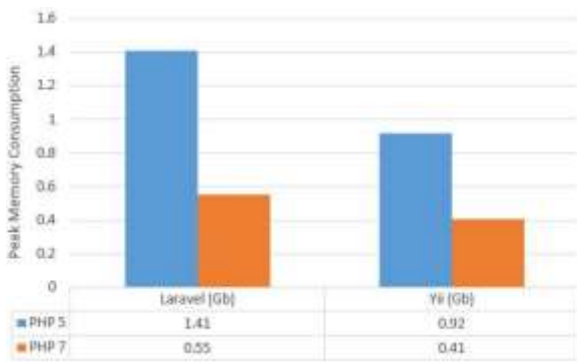
Fig. 9 Average Comparison of peak memory consumption between PHP 5 and PHP 7 using employee's table

4.1.Throughput Comparison

In this section, we want to show the throughput comparison between Laravel and Yii2 framework implemented in PHP version 5 and PHP version 7. Figure 10 shows the throughput average in records per second where higher result number has the better performance. From this testing, Laravel has better performance than Yii2. Laravel and Yii2 perform slightly different in same PHP version, whether version 5 and 7. But when we concern on throughput different between two versions of PHP, we found that implementing those frameworks in PHP version 7 give much better throughput result than in PHP version 5.
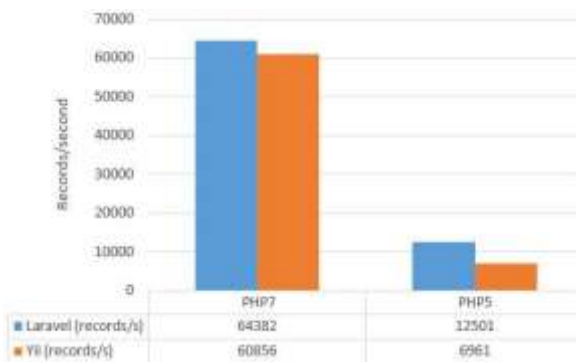

Fig. 10 Throughput comparison in records/second between PHP version 5 and PHP version 7

V. CONCLUSION

The performance comparison between Laravel and Yii2 framework that we have done in this paper was based on execution time, peak memory usage and throughput metrics. The resulting test shows that in execution time and throughput measurement, Laravel give better result with a slightly different number. Another comparison with different metrics and framework processes also allows the user get different conclusion results. This result is limited only for comparison which explained in the previous sections.

REFERENCES

[1] PHP Group. (2007). PHP Hypertext Preprocessor (2008). *URL http://www. php. net*.

[2] W3Techs. Usage of server-side programming languages for websites. URL http://w3techs.com/technologies/overview/programmin p_language/all.

[3] T. Catarci, M.F. Costabile, S. Levialdi, S., & C. Batini. (1997). Visual query systems for databases: A survey. Journal of Visual Languages & Computing, 8(2), 215-260.

[4] **B.** Shneiderman (1997, January). Direct manipulation for comprehensible, predictable and controllable user interfaces. In Proceedings of the 2nd international conference on Intelligent user interfaces (pp. 33-39). ACM.

[5] J. Dahse (2010, August). RIPS-A static source code analyser for vulnerabilities in PHP scripts. In Seminar Work (Seminer Çalismasi). Horst Görtz Institute Ruhr-University Bochum.

[6] D.K. Liefer, & S.K. Ziegler. (2008). PHP Vulnerabilities in Web Servers.

[7] M.E. Fayad, D. C. Schmidt, & R.E. Johnson. (1999). Implementing application frameworks: object-oriented frameworks at work. John Wiley & Sons, Inc.

[8] M.E. Fayad, D. C. Schmidt, & R.E. Johnson. (1999). Building application frameworks: object-oriented foundations of framework design. John Wiley & Sons, Inc.

[9] M.E. Fayad, & D. C. Schmidt (1997). Object-oriented application frameworks. Communications of the ACM, 40(10), 32-38.

[10] R.E. Johnson & B. Foote (1988). Designing reusable classes. Journal of object-oriented programming, 1(2), 22-35.

[11] M. Fayad & D.C Schmidt (1997). Object-oriented application frameworks. Communications of the ACM, 40(10), 32-38.

[12] D.C. Schmid, A. Gokhale, & B. Natarajan (2004). Frameworks: Why they are important and how to apply them effectively. ACM Queue magazine, 2(5).

[13] A. Zurkiewicz, A., & M. Milosz. SELECTING A PHP FRAMEWORK FOR A WEB APPLICATION PROJECT—THE METHOD AND CASE STUDY.

[14] Zend Technologies, "Turbo Charging the Web with PHP7".2017.*URL http://www.zend.com/en/resources/php7_infographic.*

[15] R. Das & L.P Saikia (2016). Comparison of Procedural PHP with Codeigniter and Laravel Framework. International Journal of Current Trends in Engineering & Research, 2(6).

[16] J. Samra. (2015). Comparing Performance of Plain PHP and Four of Its Popular Frameworks.

[17] J.T. Pleva. PHP: Hypertext Preprocessor.

[18] L.Lancor & S. Katha (2013, March). Analyzing PHP frameworks for use in a project-based software engineering course. In Proceeding of the 44th ACM technical symposium on Computer science education (pp. 519-524). ACM.

[19] "Framework Technology Web Usage Statistics,". *URL http://trends.builtwith.com/framework.*

[20] M. Surguy (2016). History of Laravel PHP framework, Eloquence emerging. Maks Surguy's blog on PHP and Laravel.

[21] M. Bean (2015). Laravel 5 Essentials. Packt Publishing Ltd.

[22] J. Deacon (2009). Model-view-controller (mvc) architecture. Online][Citado em: 10 de março de 2006.] http://www. jdl. co. uk/briefings/MVC. pdf.

[23] Y. Supporters, "Yii Features", 2016. *URL http://www.yiiframework.com/features/*.

[24] Apache Software Foundation. 2017. *URL http://www.apache.org*.

[25] Barry vd. Heuvel, Laravel Debugbar (Integrates PHP Debug Bar), 2013, GitHub Repository, https://github.com/barryvdh/laravel-debugbar/.

[26] Giuseppe Maxia, Test-db, 2015, GitHub Repository, https://github.com/datacharmer/test_db/.