International Journal of Communication & Information Technology (CommIT) http://msi.binus.ac.id/commit/
Vol. 7 No. 1 May 2013, pp. 28-39

WEB APPLICATION SURVEY PERFORMANCE EVALUATION PT. KALBE MORINAGA INDONESIA

Hantze Sudarma[1]; Tri Djoko Wahjono[2]
Information Technology Department, School of Computer Science, Bina Nusantara University
Jln. K. H. Syahdan 9, Jakarta 11480, Indonesia
[1]hantze@binus.edu, [2]tri_djoko_wahjono@yahoo.com

Abstract: Research on this case study aims to evaluate the web application survey PT. Kalbe Morinaga Indonesia. This case study was done because of some performance issue on the application. The problems related to the ability of software to speed data processing and reporting. Research was done on the performance of software and was combined using the gray box, that is, the evaluation of the test and evaluation of the code of the network itself. In addition, it was used for testing software for test automation. Automation tests were done as much as several times with a different number of virtual users. Results obtained were analyzed so as to be used as a reference for the results which pages have problems with performance issues, and were tested by using a solution which then were tested again on that page and comparing the results. The final results of the analysis were compared with the results of the first test and then the results of the analysis and recommendations for improvement can be used by application developers in the future.

Keywords: Evaluation; Performance Test Application; User Satisfaction; Automation Test

INTRODUCTION

The rapid advancement of technology in today's modern world gives a tremendous impact on many fields. One of the effects is felt by the practical management in the field of human resources. When first used in large organization, [1] showed that the software used in the resource section has been used in various types of organizations for all sizes. Software has been used in every function of the resources, both from the recruitment function to the strategic management function [2].

PT Kalbe Morinaga Indonesia uses the software in the form of a website that can be accessed internally by all employees. The reason for choosing the web to do surveys because it is more convenient for respondents in filling the form rather than using paper and pencil in the past due to privacy by using a personal computer [3]. The survey is used by PT Kalbe Morinaga Indonesia aims to determine the level of satisfaction of co-workers, bosses and companies. The results of the survey conducted, will be calculated and used as a component in the calculation of Key Performance Indicators (KPI) of each employee.

Software to conduct internal surveys among employees has been used for two years since 2010. Parties HR PT Kalbe Morinaga Indonesia feel that there is a lack of information provision of the software, especially in terms of providing reports of perceived inadequate. In addition there are a few complaints about the slow performance of the web-based applications, especially in the provision of reports and the lack of familiarity of the web user interface in terms of the software itself in the delivery of information.

From the discussion, it is necessary the holding of a case study to look at the gap between the level of quality of service provided by the software used, determine the level of satisfaction of users of the software with the software itself using multivariate control charts. Thus it will be able to know the circumstances that exist at the moment, in order to do repairs on parts needed or improved services as well as maintaining the good.

METHOD

Case studies were carried out based on a web-based survey application that has been used over the last two years and has led to some problems that have been identified either by the user or administrator. It was based on the research problem of supporting data collected through several means such as literature, assessment data and interviews. Data were obtained through several sources both from users and application developers.

The problems relate to the ability of software to speed data processing and reporting [4].

Research on the performance of software was done and combined using the gray box, that is, the evaluation of the test and evaluation of the code of the network itself. Then, it was used for testing software for test automation. It was used to accelerate the research and that research easier.

Automation tests were done as much as several times with a different number of virtual users. Results that obtained were analyzed so as to be used as a reference for the results which pages have problems with performance issues and tested by using a solution which were then tested again on that page and comparing the results.

The final results of the analysis were compared with the results of the first test and then the results of the analysis and recommendations for improvement can be used by application developers in the future.

Software Testing

Software testing is the process of evaluating a system or component in order to find that whether the software meets the requirements. The actual results of these activities expect results and the differences between them. In simple words, execute testing system is to identify gaps, errors or missing requirements against actual desires or requirements.

According to the standard ANSI/IEEE 1059, testing can be defined as a process of analyzing software item to detect the differences between existing conditions and the required (which defects/errors/bugs) and to evaluate the features of the software item [5].

Types of Software Testing

There are two types of ways in doing testing software [5]. They are manual testing and automation testing. Manual testing is the test is performed manually, i.e. without using automated tools or writing on it. In this type tester took over the role of the end user and test the software to identify responses that are not expected or bugs. There are different stages for manual testing such as unit testing, integration testing, system testing and user. Testers use test plans, test cases or test scenarios to test the software to ensure completeness of testing. Manual testing also includes testing the exploration as testers explore the software to identify errors in it.

While automation testing is when the tester to write scripts and use other software to test the software. Automation testing is used to re-run the test scenarios carrying out manually, quickly and repeatedly. Regardless of regression testing, test automation is also used to test the application of the load (load testing), performance (performance testing) and stress testing. This increases test coverage, increase accuracy, save time and money compared to manual testing.

Testing Methods

There are several methods for testing web [5]. Black Box Testing is testing technique which is executed without need for knowledge about the workings of the interior (code) of the application. Tester is aware of system architecture but does not have access to the source code. Usually, when doing black box testing, the tester will interact with the user interface system by providing input and check output without knowing how and where the input work are. White Box Testing is a detailed investigation of the internal logic and structure of the code. White box testing is also called glass box testing or open testing. In order to perform white box testing on the application, the tester needs to have knowledge of the internal workflow code. Tester needs to look at the source and know the unit/set of code contained in the software. Grey Box Testing is a technique to test the application by combining the white box testing and black box testing in software testing. Mastering the system always gives the tester an edge over someone with a limited knowledge of the system, both in terms of code and workflow software. Unlike black box testing, where the tester only tests the application user interface, in gray box testing, the tester has access to the documents and database design. By having this knowledge, the tester can better prepare test data and test scenarios when creating test plans.

Level of Testing

In software testing, there are several different levels during the test [5], in which each level of testing includes several different testing methodologies. Here is some level of software testing.

Functional Testing

Functional testing is testing the use of black box testing method based on testing software code based specification of software to be tested. Application will be tested to provide fill and then the results examined. Functional testing of the software is done in full, based on the needs in accordance with the standards set by the company.

There are five steps in performing functional application testing: (a) determining which parts to test in accordance with requirements, (b) determining the data that will be entered (input) when testing takes place, (c) determining the output (output) based on predetermined enter, (d) writing test scenarios and test cases execution, (e) a comparison of actual and expected results based on the test results that have been executed.

A practice test will be effective if following the steps as a standard test in all organizations. The steps provide strict standards for testing software in an organization. In the functional testing, it is identified several types of test sequences which are applied as standard testing. They are unit testing, integration testing, system testing, regression testing, acceptance testing.

Unit testing is performed by the developer before the application is submitted to the board of examiners to conduct formal application test cases. Unit testing is done by the respective app developers on any part of the code that the area has been determined. The developers use a separate test data from the test data to be used by a team of quality testers. The goal of unit testing is to isolate each part of the program and show that each of the parts of the program has no errors either in terms of functionality and based on the standards set by the company. Limitations in unit testing are not able to capture all of the bugs out of the application. This is not possible because it is not possible to evaluate every execution path in software. Developers can use limited number of scenarios and test data to verify the source code. So after the developer runs out of options, then stalled because testing is usually considered to have been successful.

Integration Testing is testing the integration of systems together, whether of the code that was created by the developer as well as of the organization. This test intends to determine whether the results have been so in accordance with the earlier agreement. There are two methods of doing integration testing, namely bottom-up and top-down integration test integration test. Bottom-up integration testing begins on unit testing (testing of internal team of developers), followed by a further test both in terms of calling modules/functionality more thoroughly. Top-down integration test starts from the calling modules/functionality higher and slowly headed to the module/reduced functionality. In a good software development environment, bottom-up testing is usually not performed alternately with the top-down testing. The process of covering a wide range of test applications thoroughly based on scenarios that have been designed to prevent errors that would occur either from the user, the system itself or other technical errors.

System testing is the next level in the test and the test system as a whole. After all the components integrated, the overall application tested rigorously to see that the application has met quality standards. This type of testing is done by a special team. Testing the system is very important because this step is the first step in the Software Development Life Cycle, in which the application is examined as a whole. Applications are thoroughly tested to verify that the application meets the technical specifications and functionality. Applications must be tested in an environment that is very close to the production environment where the application will be applied. Testing of the system allows us to test, verify and validate both for business and architectural importance of the application itself.

Regression testing is done if there is some part of a software application changes and where the changes will affect several other areas of the application itself. To verify that the bug still does not produce the wrong function in some parts of the software as well as a whole. The purpose of regression testing is to ensure that such changes do not cause problems bug fixes or other errors that are found in the application. Regression testing is very important because it minimizes the gap and application errors to the changes that have been made. New test is done to verify that the changes did not affect other areas of your application. This reduces the risk of errors due to the test can be performed when changes occur without further extending the development schedule. Therefore regression testing is essential to improve the speed in verifying the application so that the application can quickly be used or marketed.

Acceptance testing can be considered as the most important type of test in test functionality. The test is performed by the quality assurance team, and the test is performed to measure the extent to which the application meets the specifications and expected application as the user desires. Quality assurance team will have a set of pre-written scenarios and test cases that will be used to test the application. The number listed in the testing of ideas can improve the accuracy of the application to be accepted by users. This test is not only intended to demonstrate simple mistakes such as spelling, errors or gaps display interface, but is also intended to show the bugs in the applications that you will produce a larger error based on the errors found in the application. By doing acceptance test on the application under test, the team can infer the extent to which the application can be used or accepted by users, especially users who are close to the production area or user software directly. Acceptance testing is divided into two stages, Alpha Testing and Beta Testing.

Alpha testing is the first phase of acceptance testing and carried out only in the area of application development team. Unit testing, integration testing and system testing are combined to perform testing alpha. During this phase there are some things that are usually tested, things like spelling errors, broken link checking, the time needed to launch applications using a machine that has the lowest specs for running applications, as well as latency issues when the application is run. Beta testing is done after alpha testing is done. In testing the beta testing of samples from the audience or user needs as the main tester of applications. Beta testing is also known as pre-release testing. The beta version of the software should ideally be distributed to many people, as a form of test programs in a real world and a preview of the results of these tests will be used as a reference basis for the release of the application. In this phase the user will use and run the application in order to provide feedback to a team of application developers. Every little mistake, both in the things that make it difficult for users both in terms of spelling errors in the application or even crashes, the user must report them so that improvements can be made. Result of user feedback should be examined and repaired by a team of developers to ensure the same problem does not arise again in the next version. The more problems arise and the more improvements are made, the higher the level of quality applications. This is because the fewer the problems that arise, meaning the higher the level acceptable to the user application. The high quality of an application is determined by the level of user satisfaction.

Non-functional Testing

Testing is conducted by the non-functional testing of applications without grounded by the functionality of the application itself. This testing involves testing of software performance levels, security, user interface and so forth. Here are some types of non-functional testing which are usually used by the wider world. They are:

Performance Testing

Performance testing is widely used to demonstrate and verify the performance issues or common problems that occur as a bottleneck. Performance testing is not intended to point out the mistakes of the application code (bugs) in the software. There are a few things that often become a problem in the software application performance degradation, such things as network delay, client side processing, database transaction processing, load balancing between servers and data rendering.

Performance testing is usually performed as one of the most important and mandatory part of the non-application testing functionality. The aspects are such as speed (both in terms of application response time, rendering and accessing the data), capacity, stability and scalability. Performance test can be either qualitative or quantitative testing activities and can be divided into sub-types of testing such as Load and Stress testing.

Load testing is a testing process that tests the behavior of software by applying the maximum load in terms of accessing applications and input data manipulation in large numbers. This can be done at normal and peak load conditions. This type of testing and the software identifies the maximum capacity during peak behavior. Usually load testing uses tools to perform automated testing. Virtual users (*VUsers*) of the automation tools defined in automated testing tools and scripts are executed to verify the load testing software. Number of users can be manipulated according to the needs, both increases and decreases.

While stress testing is a software testing process behavior under abnormal conditions. This test uses the resources that exceed the normal limits of the load software; therefore the test is referred as stress testing. The main goal is to test the software by applying the load to the system and retrieve resources commonly used software to identify the breaking point (shut down). This test can be done by randomizing port network and performing actions such as restart or shut down the network port, turn off or turn on a database with random time and run different processes in order to reduce the computer's resources such as CPU, memory or other things from the server application.

Usability Testing

Usability testing covers concepts and different definitions of usability testing of a software standpoint. Black box testing methods are used instead of the software to catch errors (bugs), but the fault of the user while using the software and finding errors in software. Usability testing can be defined into five factors of efficiency, learning (easy to learn), satisfaction (satisfactory), the software is easy to remember and easy to use. Software can be useful and work well if users have all of the above factors.

Usability testing is a requirement of software to be well received by users, because it deals with the interaction between the human systems. If end users are satisfied with the finished application, the software can be effectively used by the user. This is determined by several standard and quality of models and methods that define usability in the form of attributes and sub-attributes, as described in ISO-9126, ISO 9241-11, ISO-13407 and IEEE std.610.12 and others.

Testing Graphical User Interface (GUI) software ensures that the software is fit and proper in terms of color selection, alignment, size and other properties associated with the display software to be well received by users. Usability testing is to make sure the other side of the user interface is designed to be user friendly and easy to use, easy to remember, efficient, does not cause confusion for end users (end user). Testing the user interface is a sub part of the usability testing.

Security Testing

Security software security testing includes testing of the security to identify any gaps contained in the software. This is important because the crevice or spot any software vulnerabilities to cyber-attack from outside or from within. Every corner of the software must be considered in order not to cause social problems such as the loss of personal data, hijacking the user's identity, as well as cracking software by others to be private property.

Portability Testing

Portability testing covers software testing is done with the aim of checking whether the software can be used repeatedly (re-useable) and can be transferred to the operating system or software from another. Here are some ways to make portability testing, first is doing some experiments to install the software from one computer to another. The other is installing the software from one platform to a different platform (good operating system, browser, and so forth).

Portability testing is considered as one of the significant test of testing a system, as this type of test can be used as a point of measuring the extent to which software can accept different environments (interoperability). Differences in computer, operating system and browser become the main focus of this testing. Portability testing can be done if the software has set the ground rules that the software can be placed in different environments. This test can be performed after the testing phases such as unit testing; integration testing has been conducted on software-related and environmental constraints for testing has been done.

Challenges during Web Testing

Testing faces various challenges during testing and should be able to conduct tests for [6]: (a) browser compatibility, (b) operating system compatibility, (c) windows application compatibility where required (especially for backend testing) allows a user to specify how virtual users are involved in the testing environment i.e. either increasing users or constant users or periodic users load, (d) increasing user load, step by step is called Ramp where virtual users are increased from 0 to hundreds, where, Ramp test is the test which uses escalating numbers of users over a given time frame to determine the maximum number of users the webserver can accommodate before producing error messages, (e) constant user load maintains specified user load at all time, periodic user load tends to increase and decrease the user load from time to time.

Websites offer new challenges to developers, as well as testers. Scalability and performance are two areas that are significant in the e-commerce and web applications space. However, when new technology is used to make web applications perform well and scalable, new testing methodologies have to be created along with those technologies. Performance is critical, and based on a study from the Newport Group, more than half the recently deployed transaction-based web applications did not meet expectations for how many simultaneous users their applications could handle [7].

Throughput

Throughput basically has a different definition with bandwidth. Bandwidth is the amount of the amount of data flowing through the network in size bits per second, while the throughput is the average amount of packets that are sent over the network. The data referred to here can be either physical or logical link, or routed over a particular network node.

On the network, throughput limited by factors such as the network protocol used, the ability of routers and switches, as well as the type of cable and fiber optic ethernet. Throughput in wireless networks is affected also by the network adapter found on the client system.

Web Application Performance Tools (WAPT)

Web application performance tools (WAPT) is used to test the web application and the interface of the web itself. This tool is used to load and stress testing on web applications, web sites, web servers and interfaces.WAPT tend to simulate virtual users who will repeat either recorded a URL or a specific URL and allows the user to specify how many times or iterations that virtual users will have to redo the recording URL. Thus, this tool is useful to examine the performance bottlenecks and leaks on the website or web application being tested.

RESULTS AND DISCUSSION

The following is a discussion of the results and subsequent evaluation.

Interview Results

Interview conducted on two surveys web developerPT. Kalbe Morinaga Indonesia with the same question in the questionnaire, the interview lasted approximately 20 minutes for each person. Interviews were conducted to collect data while checking data from the questionnaire as a form of triangulation methods were performed.

As for the interviews with administrators or people who use the system directly, from the head of Human Resource and Development Division PT. Kalbe Morinaga Indonesia. He said that there are many features that are less than satisfactory, especially in the withdrawal of the report. This affects the performance of the HR division at the end of the year because of the results of this survey report will be used to calculate KPI (Key Performance Index) of each employee. On the other hand, it is still difficult to synchronize employee data as it occurs during the data redundancy of employees in the survey web applications with Pro-INT application.

Interviews showed the difficulty of the developer is to meet user expectations, especially in terms of responsiveness and assurance web applications survey PT. Kalbe Morinaga Indonesia. Assume that the vulnerable application developers in providing quick response to the user over the problems caused by the Operating System that is used by PT. Kalbe Morinaga Indonesia, namely Windows Server 2008, while on the other side of the survey web applications using PHP technology with XAMPP web server. However, because of the limitations that exist due to regulatory changes Server Operating System and Web server survey of web applications is not possible. This means that the assessment will be carried out only from the technical area of application.

Analysis of Non-functional

To perform this analysis, it was done the analysis of the structure of the application in the server PT Kalbe Morinaga Indonesia. Data on the architecture of a web survey PT Kalbe Morinaga Indonesia were obtained by direct interview to IT supervisor PT Kalbe Morinaga Indonesia. Web surveys can only be accessed internally via the link http://10.175.12.25/kmi, where the server is responsible for storing application file. This server is owned by IT division tasked to serve requests from the user. As for the database, it is stored on the server with the IP 10.175.11.45. This server is a server owned by HR division, where no random people have access into this server. The purpose of separation between the application server with the database is that the division of the workload as well as the separation of access rights. Because of web survey under the auspices of the HR division, then the data from the HR division should not be carelessly opened to a party or other divisions.

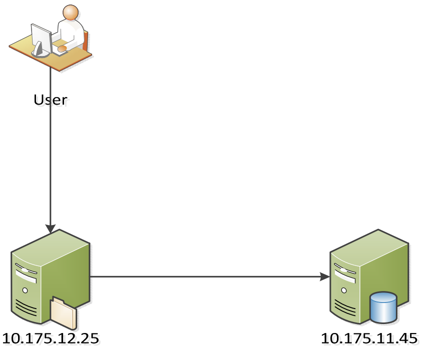


Fig. 1: Web Application Architecture Survey PT Kalbe Morinaga Indonesia

The following are the specifications of the application server used by web survey PT Kalbe Morinaga Indonesia.

Table 1: Application Server Specifications

Processor	Intel Xeon Quad Core E3-1220 (3.10 GHz, 80W, 8M)
Memory	2 x DDR3 2 GB SATA RAID
Software	Windows Server 2008, XAMPP, Pro-INT

Table 2: Specifications Database Server

Processor	Intel Dual Core G6950 (2.80GHz, 73W, 3MB, 1066)
Memory	2 x DDR3 1 GB SATA RAID
Software	Windows Server 2008, XAMPP

Basically, PT Kalbe Morinaga Indonesia's server criteria meet existing standards; this is because PT Kalbe Morinaga Indonesia has designed both the network and the server. Which can be further consideration by PT Kalbe Morinaga Indonesia is Operating System used to run a web survey application with the PHP platform. PHP-based application that will improve performance when using Linux web server uses the engine Nginx web server [8]. 4:21 look at the information table that used web server is XAMPP, of course it can be used as fill to make changes on the server web server PT Kalbe Morinaga Indonesia. However, due to regulatory restrictions, this can't be done.

Performance Analysis of Web Survey

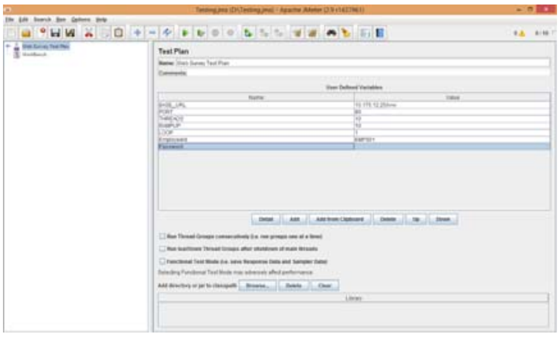
To perform performance analysis tools to be used in the form of open source automation software called Jmeter. This software supports virtualization based on the number of users accessing virtual web, web pages per user, as well as the number of users accessing it themselves. Software Jmeter will be placed on a single computer at PT. Kalbe Morinaga Indonesia, to capture quantities predetermined variables. As these variables is the minimum amount of time accessing per page, the maximum time, average time, the amount of throughput, as well as the standard deviation.

To start the performance test, the predetermined value constants that apply when performing the test. The other constant is the constant link between the primary (base url) of the web you want to be tested, the main port of the link, the total virtual threads or user accessing, loading RAM, accessing the number per user, as well as other additional variables that will be used to test try.

Table 3: Specifications Database Server

Processor	Intel Core 2 Duo T5900 (2.20GHz, 35W, 2MB)
Memory	2 x DDR2 1 GB SATA RAID
Operating System	Windows 8

Fig. 2 shows the testing will be done by accessing the url 10.175.12.25/kmi port 80. Number of virtual users accessing user is 10, the amount of RAM is 10 times the loading and the amount per user is accessing at one time. While the rest are the variables that will be used to perform authentication and authorization when accessing to all web pages.


Testing.jmx (D:\Testing.jmx) - Apache JMeter (2.9 r1437961)

Test Plan

Name: Web Survey Test Plan

Comments:

User Defined Variables

Name:	Value
BASE_URL	10.175.12.25/kmi
PORT	80
THREADS	10
RAMPUP	10
LOOP	1
Employeeid	EMP001
Password	

☐ Run Thread Groups consecutively (i.e. run groups one at a time)

☐ Run tearDown Thread Groups after shutdown of main threads

☐ Functional Test Mode (i.e. save Response Data and Sampler Data)

Selecting Functional Test Mode may adversely affect performance.

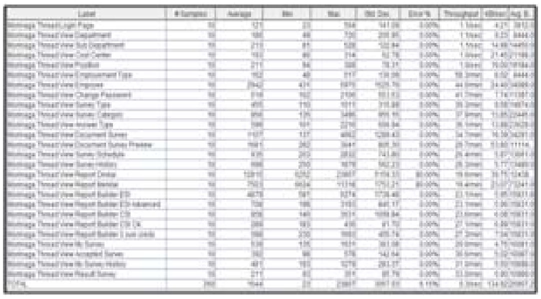
Add directory or jar to classpath

Fig. 2: The initial view Jmeter with User Defined Variables

Test Drive with 10 Virtual

After the initialization of the initial value, the test plan will be carried out on any web pages contained in the PT. Kalbe Morinaga Indonesia. All these web pages will be accessed and incorporated the input according to the needs and checked how much the value of the variables that have been previously determined. If it had been obtained, the next is to find the fundamental problems of the pages that had large processing time (based on the criteria of mean other pages).

Fig. 3 is the result of a test plan summary that has been done by simulating 10 virtual users. The amount of time specified in the image above in milliseconds. From the picture above we can see that page report assessed, assess and report builder ESI page shows the average response time of the most high, 12 815, 7,503 and 4,879 milliseconds. This is consistent with the conclusions drawn from the questionnaire survey user that the web application had weakness on the web and also that the response of the difficulty the user to pull the report from the website. In addition, the results of the test plan summary shows that the highest error rate is shown also by page report assessed and assessed, which amounted to 80%. This is similar to the results of trials that have been performed functionality, where a lot of errors that were arrested when tested blind for inputting data into the web page to draw the report. 10 images obtained from the graph of response time performance.



Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	KB/sec	Avg. B.
Morinaga Thread Login Page	10	171	23	554	141.06	0.00%	1.1/sec	4.21	[illegible]
Morinaga Thread View Department	10	186	48	720	200.85	0.00%	1.1/sec	9.23	8444.0
Morinaga Thread View Sub Department	10	215	81	528	120.84	0.00%	1.1/sec	14.68	[illegible]
Morinaga Thread View Cost Center	10	153	80	314	70.78	0.00%	1.0/sec	21.45	[illegible]
Morinaga Thread View Position	10	211	94	388	78.17	0.00%	1.0/sec	76.09	[illegible]
Morinaga Thread View Employment Type	10	152	48	317	130.06	0.00%	58.3/min	8.02	8444.0
Morinaga Thread View Employee	10	2942	431	6975	1925.76	0.00%	44.0/min	24.40	[illegible]
Morinaga Thread View Change Password	10	518	102	2106	651.63	0.00%	41.7/min	1.16	[illegible]
Morinaga Thread View Survey Type	10	455	110	1011	315.88	0.00%	39.3/min	8.58	[illegible]
Morinaga Thread View Survey Category	10	856	125	3486	955.15	0.00%	37.6/min	13.85	[illegible]
Morinaga Thread View Answer Type	10	596	101	2216	656.84	0.00%	36.1/min	13.88	[illegible]
Morinaga Thread View Document Survey	10	1127	117	4062	1288.43	0.00%	34.7/min	16.39	[illegible]
Morinaga Thread View Document Survey Preview	10	1661	262	3441	805.30	0.00%	29.7/min	13.60	[illegible]
Morinaga Thread View Survey Schedule	10	635	203	2832	743.80	0.00%	29.4/min	1.87	[illegible]
Morinaga Thread View Survey History	10	996	250	1878	562.23	0.00%	28.3/min	1.77	[illegible]
Morinaga Thread View Report Dinilai	10	12815	8252	23667	5104.33	80.00%	19.8/min	10.71	[illegible]
Morinaga Thread View Report Menilai	10	7503	6634	11116	1751.21	80.00%	19.4/min	21.17	[illegible]
Morinaga Thread View Report Builder ESI	10	4879	597	9274	2728.46	0.00%	23.1/min	1.85	[illegible]
Morinaga Thread View Report Builder ESI Advanced	10	739	186	3193	846.17	0.00%	23.1/min	1.86	[illegible]
Morinaga Thread View Report Builder CSI	10	878	140	3517	1059.84	0.00%	23.6/min	4.08	[illegible]
Morinaga Thread View Report Builder CSI CK	10	269	183	416	61.70	0.00%	27.1/min	4.89	[illegible]
Morinaga Thread View Report Builder 1 sum credit	10	598	230	1665	455.74	0.00%	27.2/min	7.34	[illegible]
Morinaga Thread View My Survey	10	536	135	1631	361.08	0.00%	28.0/min	4.75	[illegible]
Morinaga Thread View Accepted Survey	10	392	98	576	142.94	0.00%	30.5/min	1.52	[illegible]
Morinaga Thread View My Survey History	10	421	183	1276	283.77	0.00%	31.0/min	1.93	[illegible]
Morinaga Thread View Result Survey	10	271	83	311	85.79	0.00%	33.0/min	1.93	[illegible]
TOTAL	260	1644	23	23667	[illegible]	6.15%	8.3/sec	114.62	[illegible]

Fig. 3: Summary results of the Test Plan

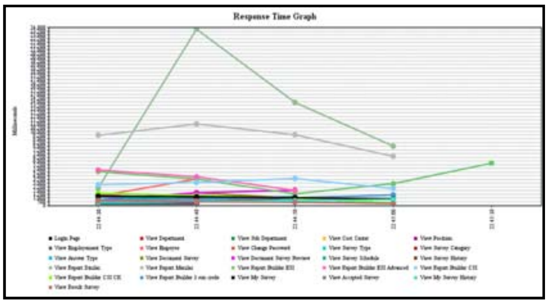


Fig. 4: Response Time Graph

Fig.4 shows the time taken by Jmeter to do a page access. From the graph, it can be seen that the time required to complete all assigned tasks indicated by the ESI report page builder. While the page with the largest total time when a request is page report assessing the report assessed and ranked second most time to make the request. Span of time between the hours of 22:44:30 to 22:45:10 pm.

Fig. 4 is a bar chart to show the average time of each virtual user to access web pages. Page report assessed, assess and report builder ESI page shows the average response time of web pages were the highest compared to the other.

Fig. 5 shows the data transfer speed of accessing each web page. The picture above shows the magnitude of the transfer rate required from page report evaluated and assessed. From the graph it can be seen that the transfer speed of page report assessed, assess and ESI ranks lowest in data transfer speed. This could indicate that the slow processing time compared to the pages of other pages.

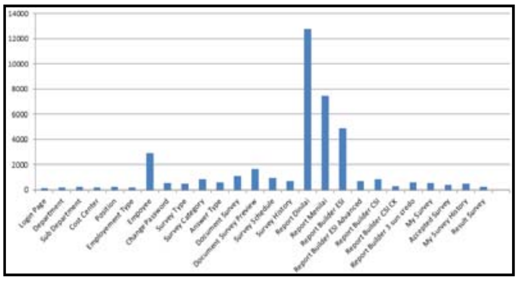


Fig.5: Aggregate Average Rate Graph

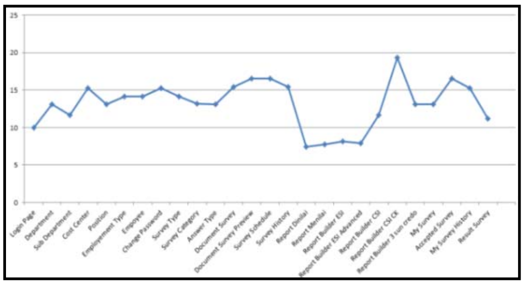


Fig. 6: Total Transfer Rate Graph

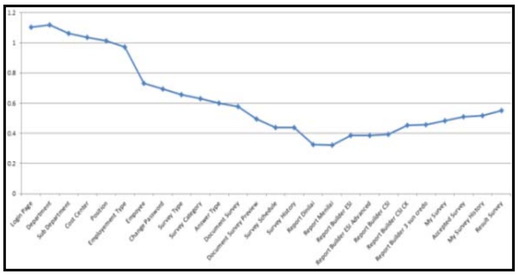


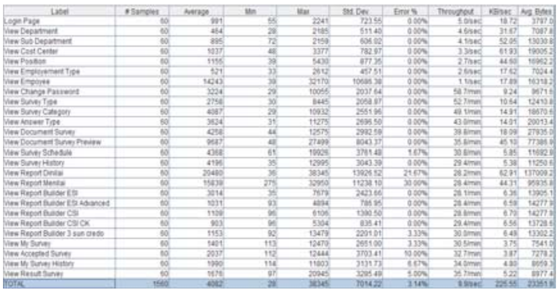
Fig. 7: Total ThroughputGraph

Fig. 7 shows the amount of throughput or average rate of successful data transmission from each web page. The higher the average speed, the better, but you look back that page report has assessed and assess data transmission speeds slower than other pages. It can be concluded that the report assessed the page and judge did experience a longer processing time than other pages.

Test Drive with 60 Virtual

The conclusion that can be drawn from simulation experiments testing the performance of web surveys PT. Kalbe Morinaga Indonesia is the need of holding an improvement on a web page report assessed, assess and report builder ESI. But to be sure, will be held back by the number of threads simulation or virtual user for 60 virtual users as a form of realization of the streak when accessing the web survey open to all employees.

Fig. 8 is a picture that shows the actual results of performance tests by the number of virtual users by 60 employees.



Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	KB/sec	Avg. Bytes
Login Page	60	991	55	2241	723.55	0.00%	5.0/sec	18.72	3797.0
View Department	60	464	28	2185	511.40	0.00%	4.6/sec	31.67	7087.8
View Sub Department	60	895	72	2159	606.92	0.00%	4.1/sec	52.05	13030.8
View Cost Center	60	1037	48	3377	782.97	0.00%	3.3/sec	61.93	19005.2
View Position	60	1155	39	5430	877.35	0.00%	2.7/sec	44.60	16962.2
View Employement Type	60	521	33	2612	457.51	0.00%	2.6/sec	17.62	7024.4
View Empoyee	60	14243	39	32170	10686.38	0.00%	1.1/sec	17.89	16318.2
View Change Password	60	3224	29	10055	2037.64	0.00%	58.7/min	9.24	9671.5
View Survey Type	60	2758	30	8445	2058.97	0.00%	52.7/min	10.64	12410.8
View Survey Category	60	4087	29	10932	2551.96	0.00%	49.1/min	14.91	18670.6
View Answer Type	60	3624	31	11275	2696.50	0.00%	43.0/min	14.01	20013.4
View Document Survey	60	4258	44	12575	2992.58	0.00%	38.6/min	18.09	27935.0
View Document Survey Preview	60	9687	48	27499	8043.37	0.00%	35.8/min	45.10	77386.9
View Survey Schedule	60	4368	61	19926	3781.48	1.67%	30.8/min	5.85	11692.8
View Survey History	60	4196	35	12995	3043.39	0.00%	29.4/min	5.38	11250.6
View Report Dinilai	60	20480	36	38345	13926.52	21.67%	28.2/min	62.91	137009.2
View Report Menilai	60	15839	275	32950	11238.10	30.00%	28.4/min	44.31	95935.8
View Report Builder ESI	60	3014	35	7679	2423.66	0.00%	28.1/min	6.36	13905.1
View Report Builder ESI Advanced	60	1031	93	4894	786.95	0.00%	28.4/min	6.59	14277.9
View Report Builder CSI	60	1109	96	6106	1390.50	0.00%	28.8/min	6.70	14277.9
View Report Builder CSI CK	60	903	96	5304	835.41	0.00%	29.4/min	6.56	13728.6
View Report Builder 3 sun credo	60	1153	92	13479	2201.01	3.33%	30.0/min	6.48	13302.2
View My Survey	60	1401	113	12470	2651.00	3.33%	30.5/min	3.75	7541.0
View Accepted Survey	60	2037	112	12444	3703.41	10.00%	32.7/min	3.87	7278.2
View My Survey History	60	1990	114	11803	3131.73	6.67%	34.0/min	4.80	8659.3
View Result Survey	60	1676	97	20945	3295.49	5.00%	35.7/min	5.22	8977.4
TOTAL	1560	4062	28	38345	7014.22	3.14%	9.9/sec	225.55	23351.9

Fig. 8: Results of Test Plan Summary 60 Threads

Fig. 8 is the result of a test plan summary of the number of 60 virtual users. From the picture above we can see that page report assessed, assess and employee page shows the average response time of the most high, 20 480, 15 839 and 14 243 milliseconds. This result is slightly different from the results of performance tests with 10 virtual users, where the results shown by the test web page builder ESI report ranks the third highest.

Fig. 9 shows the time taken by Jmeter to do a page access. From the graph, we can see that the page with the largest total time when a request is assessed and a report page report assessing the highest ranks second time to make the request.

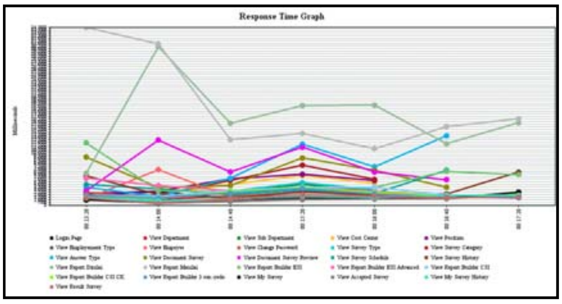


Fig. 9: Response Time Graph 60 Threads

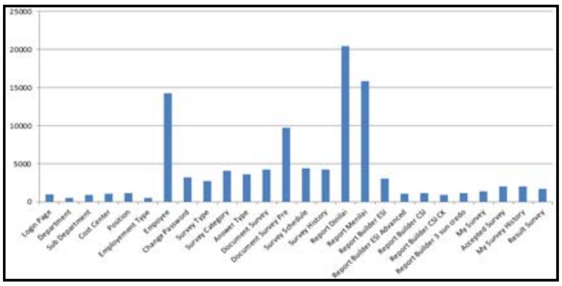


Fig. 10: Aggregate Average Rate 60 Threads Graph

Fig. 10 is a bar chart to show the average time of each virtual user to access web pages. Page report assessed and judging shows the average response time of web pages were the highest compared to the other. It is the same with the same experiment using 10 virtual users, the difference lies only in the results page employee.

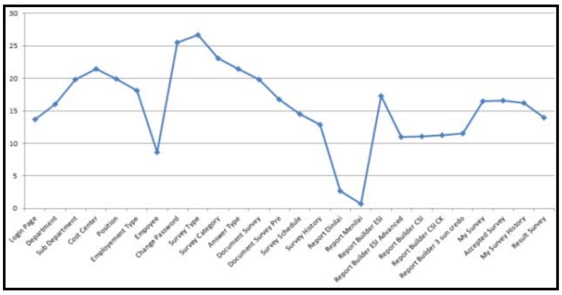


Fig. 11: Total Transfer Rate 60 Threads Graph

Fig. 11 shows the data transfer speed of accessing each web page. The picture above shows the magnitude of the transfer rate required from page report evaluated and assessed. From the graph it can be seen that the transfer speed of page report assessed, assess and Employee ranks lowest in data transfer speed. This could indicate that the slow processing time compared to the pages of other pages.

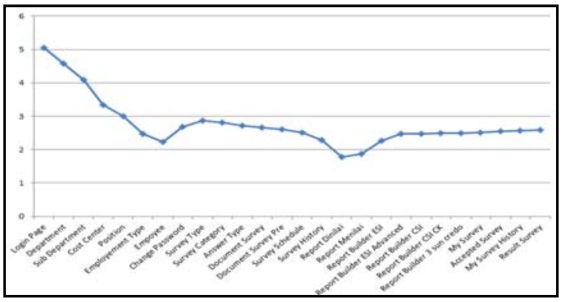
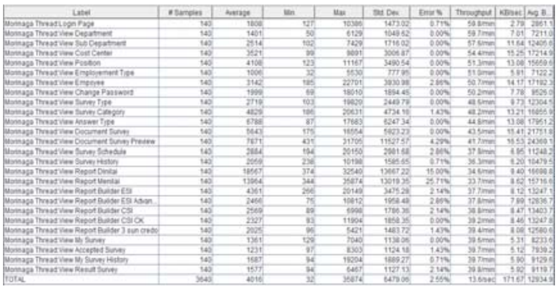


Fig. 12: Total Throughput 140 Threads Graph

Fig. 12 shows the amount of throughput or average success rate of data transmission from each web page. From the experimental results, it can be seen that the results of the experiment did not experience a significant difference, both from the results of 10 trials with 60 virtual users and virtual users. It can be concluded that the report assessed the page and judge did experience a longer processing time than other pages.

Test Drive with 140 Virtual

Fig. 13 is a picture that shows the actual results of performance tests with virtual user number of 140 employees.



Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	KB/sec	Avg. B...
Morinaga Thread:Login Page	140	1808	127	10386	1473.02	0.71%	59.8/min	2.79	2861.1
Morinaga Thread:View Department	140	1401	50	6129	1049.62	0.00%	59.7/min	7.01	7211.0
Morinaga Thread:View Sub Department	140	2514	102	7429	1716.02	0.00%	57.6/min	11.64	12405.6
Morinaga Thread:View Cost Center	140	3521	99	9891	3006.87	0.00%	54.4/min	15.25	17214.9
Morinaga Thread:View Position	140	4108	123	11167	3490.54	0.00%	51.3/min	13.08	15659.6
Morinaga Thread:View Employement Type	140	1006	32	5530	777.95	0.00%	51.0/min	5.81	7122.2
Morinaga Thread:View Empoyee	140	3142	185	22701	3930.98	2.86%	50.7/min	14.17	17182.3
Morinaga Thread:View Change Password	140	1999	69	18010	1894.45	0.00%	50.2/min	7.78	9526.0
Morinaga Thread:View Survey Type	140	2719	103	19829	2449.79	0.00%	48.6/min	9.73	12304.5
Morinaga Thread:View Survey Category	140	4829	186	20631	4734.16	1.43%	48.2/min	13.21	16855.9
Morinaga Thread:View Answer Type	140	6788	87	17683	6247.34	0.00%	44.8/min	13.08	17951.2
Morinaga Thread:View Document Survey	140	5643	175	16554	5923.23	0.00%	43.5/min	15.41	21751.8
Morinaga Thread:View Document Survey Preview	140	7671	431	31705	11527.57	4.29%	41.7/min	16.53	24369.1
Morinaga Thread:View Survey Schedule	140	2884	194	20150	2981.68	2.86%	37.8/min	6.95	11248.2
Morinaga Thread:View Survey History	140	2059	236	10198	1585.65	0.71%	36.3/min	6.20	10479.5
Morinaga Thread:View Report Dinilai	140	18567	374	32540	13667.22	15.00%	34.6/min	9.40	16698.8
Morinaga Thread:View Report Menilai	140	13964	344	35874	13019.35	25.71%	33.7/min	8.62	15716.6
Morinaga Thread:View Report Builder ESI	140	4361	266	20149	3475.28	2.14%	37.7/min	8.12	13247.1
Morinaga Thread:View Report Builder ESI Advan...	140	2466	75	10812	1958.49	2.86%	37.8/min	7.99	12836.7
Morinaga Thread:View Report Builder CSI	140	2569	89	6998	1786.36	2.14%	38.8/min	8.47	13403.7
Morinaga Thread:View Report Builder CSI CK	140	2327	93	11904	1858.35	0.00%	39.2/min	8.46	13247.8
Morinaga Thread:View Report Builder 3 sun credo	140	2025	96	5421	1483.72	1.43%	39.4/min	8.08	12580.6
Morinaga Thread:View My Survey	140	1361	129	7040	1138.06	0.00%	39.6/min	5.31	8233.6
Morinaga Thread:View Accepted Survey	140	1231	97	8303	1124.18	1.43%	39.7/min	5.12	7939.2
Morinaga Thread:View My Survey History	140	1687	94	19204	1889.27	0.71%	39.7/min	5.90	9129.6
Morinaga Thread:View Result Survey	140	1577	94	6467	1127.13	2.14%	39.8/min	5.92	9119.7
TOTAL	3640	4016	32	35874	6479.06	2.55%	13.6/sec	171.67	12934.9

Fig. 13: Summary results of the Test Plan 140 Threads

Fig. 13 is the result of a test plan summary of the number of 140 virtual users. From the picture above we can see that page report assessed, assess and document page preview survey shows the average response time of the most high, 13,667, 13,019 and 11,527 milliseconds. This result is slightly different from the results of performance tests with 10 virtual users, where the results shown by the test web page builder ESI report ranks the third highest. Results of this test plan summary also shows that the highest error rate is shown also by page report assessed and assessed, which amounted to 15% and 25.71%. This is similar to the results of the test trial performance with 10 virtual users and 60 virtual users.

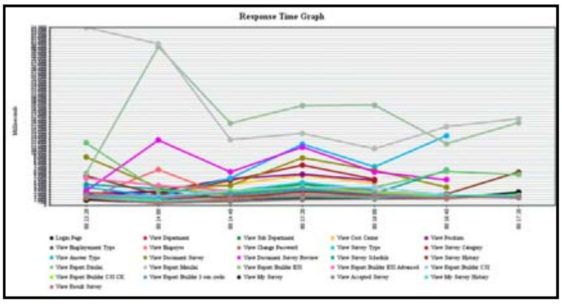


Fig. 14: Response Time Graph 140 Threads

Fig. 14 shows the time taken by Jmeter to do a page access. From the graph, we can see that the page with the largest total time when a request is assessed and a report page report assessing the highest ranks second time to make the request. While the ESI report pages are at the same position when doing yard completion to report assessed, assess, survey history, login pages, report builder CSI and report builder CSI CK.

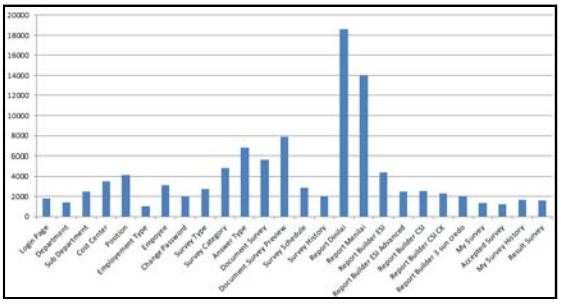


Fig. 15: Aggregate Average Rate Graph
140 Threads

Fig. 15 is a bar chart to show the average time of each virtual user to access web pages. Page report assessed and judging shows the average response time of web pages were the highest compared to the other. It is the same with the same experiment using 10 virtual users and 60 virtual users.

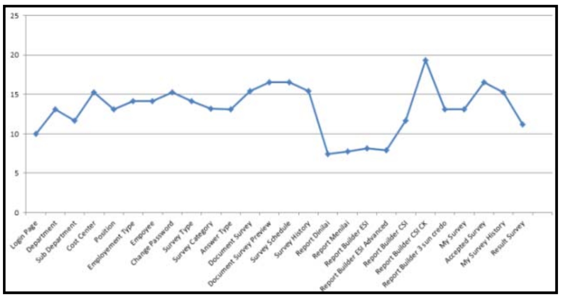


Fig. 16: Total Transfer Rate 140 Threads Graph

Fig. 16 shows the data transfer speed of accessing each web page. The picture above shows the magnitude of the transfer rate required from page report evaluated and assessed. From the graph it can be seen that the transfer speed of page report assessed, assess and ESI ranks lowest in data transfer speed. This could indicate that the slow processing time compared to the pages of other pages.

Fig. 17 shows the amount of throughput or average rate of successful data transmission from each web page. From the experimental results, it can be seen that the results of the experiment did not experience a significant difference, both from the results of trials with 10 virtual users, 60 users and 140 virtual virtual user. It can be concluded that the report assessed the page and judge did experience a longer processing time than other pages.

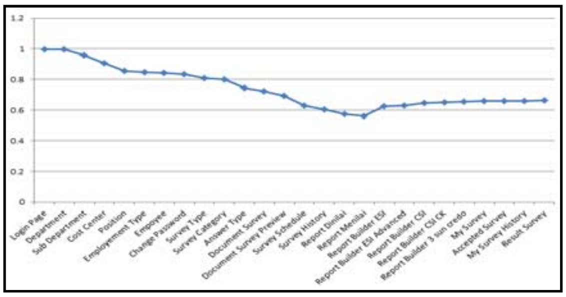


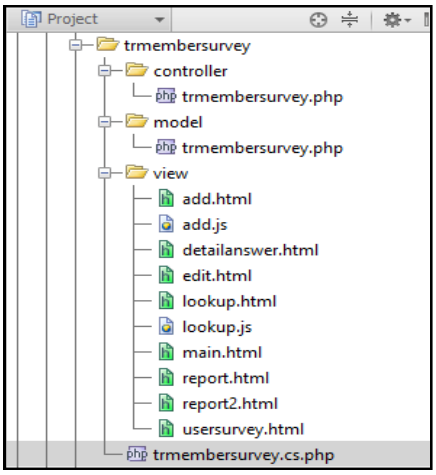
Fig. 17: Total Throughput 140 Threads Graph

Then the conclusions of any page on the web survey PT. Kalbe Morinaga Indonesia is still the same as the virtual experiments with 10 users and 60 virtual users. Therefore, it needs further assessment using the code search method and a database of applications.

Code Analysis Application

The assessment will focus on the application code page report assessed and considered, because the analysis is based on performance tests and analysis of user satisfaction questionnaire both pages of the most cause problems and deserves to be tested in white box testing.

From the analysis of the code, it was found that the framework for application development using MVC (Model, View, Controller) and an internal framework of the developer. Developers do not use existing frameworks or open source framework for building web applications survey PT. Kalbe Morinaga Indonesia.



```
Project
trmembersurvey
├─ controller
│  └─ trmembersurvey.php
├─ model
│  └─ trmembersurvey.php
├─ view
│  ├─ add.html
│  ├─ add.js
│  ├─ detailanswer.html
│  ├─ edit.html
│  ├─ lookup.html
│  ├─ lookup.js
│  ├─ main.html
│  ├─ report.html
│  ├─ report2.html
│  └─ usersurvey.html
└─ trmembersurvey.cs.php
```

Fig. 18: Hierarchical Structure Code

And when the test is done by using the Google Chrome browser, in Fig. 18 shows that the entire external source downloaded when opening page report assessed. In the code seen many looping and nested queries that occur at each iteration. This will slow down the processing time both in terms of the application server and database server. Because queries are sequential iteration will take a lot of resources, so it's good like this repetition is eliminated. From the code above, indicated that the query inside the loop due to the need to combine the data from one table to another table.

When traced by conducting interviews with developers, they say that this is due to the limitations of the MVC framework is not perfect. So it cannot support the query to combine data from one table into another table. It is certainly detrimental and large power consuming as separate the connection to query repeatedly will cause the weight of the server to do the processing.

Conclusions obtained from the analysis of the application code are the lack of a framework that is used to support and speed up the application process.

Database Analysis

After analyzing the survey web database PT. Kalbe Morinaga Indonesia, it is found several problems that can cause a lack of performance of the application. One is the absence of foreign key to the design of web survey database. This can cause a lack of performance in the absence of query indexing which controls placement of data on the storage (Hard Disk), so that when the search process takes place takes a little longer than even long enough to use a foreign key. Besides indexing is used in the database is MyISAM, indexing this type does not support the use of foreign keys that must be changed to the type of indexing with InnoDB.

Repair Trial Application

After analyzing the PT web survey. Kalbe Morinaga Indonesia, be concluded that the existence of problems in some web pages. Pages are pages that cause problems assessed and assess survey report, it was shown by several trials automation with 10 virtual users, 60 virtual users and 140 virtual users.

Given these results, it will be conducted trials report improvements in page survey assessed and assessed. The trial will only be limited to repairing the database and use other frameworks on the code page report surveys assessed and assessed. Repairs will start from the database by adding the relations between tables so that performance can be improved. In addition to the simulation re-coded using DooPHP framework. This is done because the framework is used to create web applications survey PT. Kalbe Morinaga inefficient in performing queries, so it will be tested with a mock-up applications developed with other frameworks. DooPHP framework refers to the use of benchmarks provided by http://DooPHP.com/benchmark, where the results of the performance comparison of some frameworks like CakePHP, Symfony, Kohana, Yii, as well as the Code Igniter PHP framework for showing that the most efficient is DooPHP.

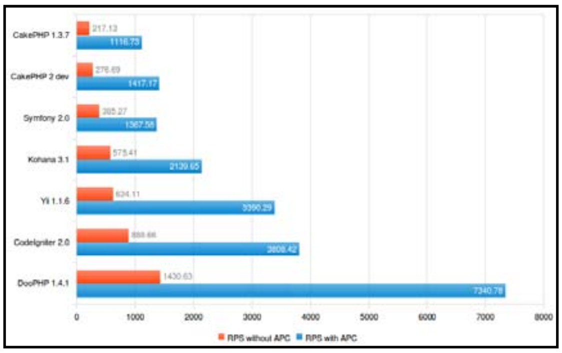
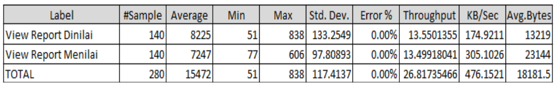


Fig. 19: Performance Comparison between Framework

Fig. 20 is the result of a test plan summary of the number of 140 virtual users with improvements in database and application framework. From the picture above we can see that page report assessed progress and assessing performance is much better than using the initial framework of web application survey. When compared to the average response time between the application and the application is a mock-up before the comparative efficiency of the use and improvement of the framework is the application of 55.70% on report pages assessed and by 48.10% on a report assessing the page.



Label	#Sample	Average	Min	Max	Std. Dev.	Error %	Throughput	KB/Sec	Avg.Bytes
View Report Dinilai	140	8225	51	838	133.2549	0.00%	13.5501355	174.9211	13219
View Report Menilai	140	7247	77	606	97.80893	0.00%	13.49918041	305.1026	23144
TOTAL	280	15472	51	838	117.4137	0.00%	26.81735466	476.1521	18181.5

Fig. 20: Summary Test Plan 140 Threads Result

In Fig. 21, it is seen a significant difference between the average response time required by older applications with a mock-up application using the framework and database DooPHP given relation. From the results obtained an increase in the response time of the application, increasing the transfer rate and increase the throughput of the application. It can certainly be used as input for the application developers for improvements in the future. By fixing the structure of the database and change the application framework, the performance improvement can be achieved so that the problem can be solved around performance.

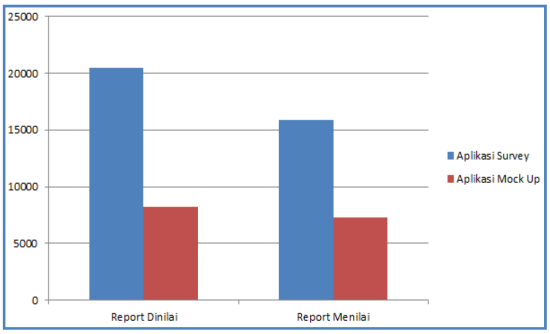


Fig. 21: Comparison of Average Response Time Results

CONCLUSION

From the analysis of case study survey web application performance evaluation PT. Kalbe Morinaga Indonesia already done then the conclusion can be drawn is as follows. From the results obtained from the analysis of the functional tests Kalbe Morinaga web survey, it is concluded that there are still some errors that appear when opening the page and enter the input into it. Then, based on the results of performance tests using test automation engineering tests showed that the pages as page report assessed and assessing performance remained below other pages. The problems surrounding the issue of slow performance is very visible on the second page, so it must be held fixes. Later, based on the results of code analysis, it was found that the limitations of the basic framework of web application survey PT. Kalbe Morinaga Indonesia to support the process of integration between the tables so that the query is a query that many repetitions overload the application server or database server. Besides database analysis, it was found that the structure of the database table from a web survey application does not use relations and indexing techniques. This causes a decrease in the performance of the query. Afterwards, based on trials with improved database and use other frameworks for making a mock-up of the application, it was found that by improving the application of the database by adding the relations between tables as well as changes in the application framework using the DooPHP efficiency improvements can be achieved. Based on these trials report increased efficiency on page assessed amounted to 55.70% and by 48.10 page report assessing%.

ACKNOWLEDGMENT

The authors would like to thank all those who have helped write and do not forget to Bina Nusantara University which has provided the opportunity to write this paper.

REFERENCES

[1] J. Schramm, “HR technology competencies: new roles for HR professional”, *SHRM Research Quarterly*, pp. 2-10, 2006.

[2] B. Rauand, & D. Feinauier, “Bringing HR into the information age why the status quo isn’t going to get us there”, *Journal of Spring*, 4(2), pp. 1-13, 2010.

[3] R. N. Graff, “Internet research: mainstream application and sustainable growth, in: advancing the Business of Research”, *Casro Annual Journal*, 2002.

[4] W. Cheung, M. K. Chang, & V. S. Lai, “Prediction of Internet and World Wide Web usage at work: a test of an extended Triandis model”, in *Decision Support Systems*, 30(1), pp. 83–100, 2000.

[5] E. W. Perry, *Effective Methods of Software Testing*, 3rd Ed., Wiley, 2006.

[6] K. Shakti, “Web Testing: Tool, Challenges and Methods”, *IJCSI International Journal of Computer Science Issues*, on March, 9(3), 2012.

[7] B. Shea, “Avoiding Scalability Shock”, *Software Testing & Quality Engineering Magazine*, pp. 42-46, on May/June 2000.

[8] J. Lee, & W. Brent, *Open Source Web Development with LAMP: Using Linux, Apache, MySQL, Perl, and PHP*, Addison-Wesley, on Dec., 2002.