Web Performance Testing of Lecturer Evaluation System (e-CEvas) applied with AJAX Technology

Azmi Abu Seman [1*], Nor Azlina Mahmud [2], Norfiza Ibrahim [3]
[1,2,3]Faculty of Computer & Mathematical Sciences, Universiti Teknologi MARA, Perlis Branch, Malaysia

*Corresponding author: *azmi384@perlis.uitm.edu.my*

Received Date: 27 August 2018
Accepted Date: 12 November 2018

ABSTRACT

Lecturer Evaluation System (e-CEvas) is used to measure lecturers' strength and weaknesses every end of semester by the students. However, the current lecturer evaluation system had many problems such as slow performance, limited interactivity, loss of states and the data transmission had redundant code that was unnecessary in each page. To overcome those stated problem, AJAX technology is used to make the lecturer evaluation system much faster, easier to use, more reliable and increase the performance. This study applied AJAX technology in the development together with Visual Studio Ultimate 2012 as the editor, ASP.NET C# language for coding and SQL Server as the database. V-model framework approach was used in developing this research study which involves requirement analysis, system design, architectural design, module design, unit testing, integration testing, system testing and acceptance testing. Microsoft Visual Studio Web Performance and Load Testing software is used to measure the performance of AJAX technology and the result shows that differences of response time proved that AJAX is powerful web development technique for programming. User Acceptance Test was carried out to measure the satisfaction of users when using e-CEvas which the result shown that most respondents are agree to all instructions in the system and the organization of information on the screens is clear.

Keywords: *AJAX, Lecturer evaluation system, web application, web performance testing*

INTRODUCTION

Lecturer's evaluation must be done to ensure the quality and performance of a lecturer reached the standard specified by the higher education institutions. The evaluation result will encourage the lecturers to improve teaching skills, subject knowledge, attitude and attendance, in order to improve student's achievement (Muzenda, 2013). Many researchers tend to agree that evaluation of teaching must encompass a variety of measures (Santiago & Benavides, 2009). Students believed that online evaluation allowed them to provide more effective and more constructive feedback (Anderson & Cain, 2005). Thus, lecturer evaluation should be supported by system that speeds up the retrieval of reliable and accurate data and result so that student and lecturer and involve department staff can rely on it and confidence when they use it.

LECTURER EVALUATION SYSTEM (E-CEVAS)

e-CEvas was implemented at Universiti Utara Malaysia (UUM) at the end of 2008. The system aims to give students the opportunity to provide feedback on the teaching and learning process that they went through during the semester. The previous version of e-CEvas had weaknesses that still need to be improve. It had many problems such as slow performance, limited interactivity, loss of states and the data transmission had redundant code that was unnecessary in each page. The search process also time consuming when the user must fill a field in page and click a button to retrieve result. Also, results from the retrieve action will take a longer time to release as the page has to load data from the server. Therefore, the results from the evaluation had took a long time to released, which was too late to make changes and took appropriate actions based on the evaluation of students. With the aims to overcome those problems, a new version of e-CEvas was develop by integrating Asynchronous JavaScript and XML (AJAX) technology to get rapid result, the requirement of less retrieval time, less vulnerability to any educator influence, cost-effectiveness, immediate reliable and accurate availability of data for analysis and reporting. The system was developed using ASP.NET C#, Visual Studio Ultimate 2012 as the editor and SQL Server as the database. V-model framework approach was used in developing this research study. The following phase was used namely: requirement analysis, system design, architectural design, module design, unit testing, integration testing, system testing and acceptance testing.

AJAX TECHNOLOGY

Web developers nowadays are looking forward in the development of web applications. They wanted to develop Web applications with more interactivity and performance improvement and, as well as user interfaces with responsive function. To meet the specified requirement, a new kind of technology used AJAX was introduced in 1990s. AJAX was a name used to a collection of technologies (Paulson, 2005) designed to enhance web application responsiveness. This collection of technologies was the combination of Dynamic HTML, CSS, HTML, as standard base presentation, asynchronous or synchronous retrieval of data using XML HttpRequest, Dynamic interaction and displays using the data exchange, Document object model (DOM) model, and execute using XML, and client-side scripting (in either Java or JavaScript/ECMAScript) to bind everything together.

AJAX allows applications to provide rich client-side interfaces and allows the browser to communicate with the web without forcing the entire page refreshes. AJAX gives better usability whenever the data changes. AJAX allows only the required part was refreshed instead of refreshing the whole page. An intermediary (AJAX engine) was introduced by AJAX application which removes the start-stop-start-stop nature of web interaction between the user and the server (Murugesan, 2007). AJAX engine, which was usually written in JavaScript and hidden, will be loaded by the browser at the beginning of each session. This engine was responsible for both communicating with the server on the user's behalf and rendering the interface the user sees. Asynchronous data retrieval from the user to the application was allowed by AJAX engine. So, the user was never staring at a blank browser window and an hourglass icon, waiting around for the server to response to any request. According to (Marchetto, 2008), AJAX Technology allows developers to realize web application which include advance, sophisticated user interactions, which are not possible with the traditional web applications.

In the traditional web application, users are presented by the browser an HTML page that includes CSS and another component. An HTTP request was sent to the server (Figure 1) when user do an action of clicking a link or submitting a form. One or more HTTP responses returns by the server contains new HTML page, together with CSS, graphics, etc.

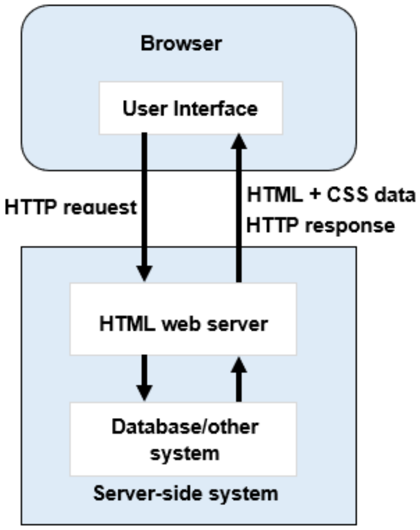


Figure 1: Traditional web application

In an AJAX application, the AJAX code will be executed behind the scene by the server according to user actions without the user ever notices any update happens to the client-side interface. The server will receive an XML HttpRequest that was send by AJAX code (Figure 2), which returns XML data. The AJAX code will update the remains active user interface without having to refresh the entire page. The user does not have to browse from page to page to do actions. Instead, AJAX application gives the ease of dynamically responding to user's actions, background interactions that are unseen by users and at the same time may be doing other actions.

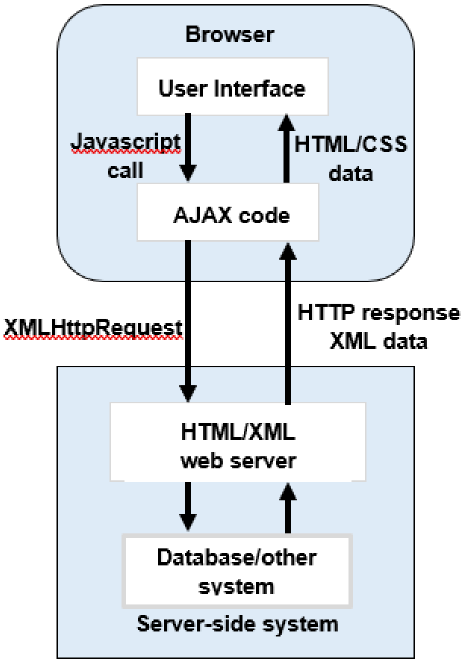
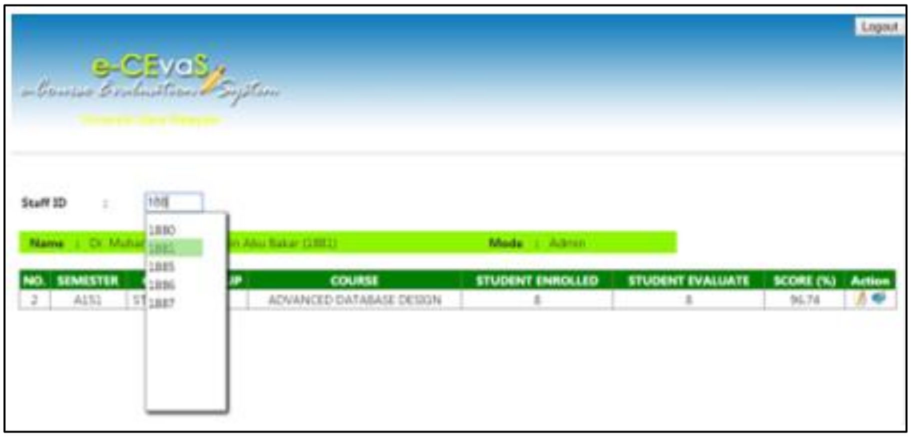


Figure 2: AJAX web application

AJAX usage in developing web applications had developed rapidly with the new browser versions that support XML HttpRequest. Social networking website that implement AJAX technology had become more popular as the fast performance of using it had increase. (Merril, 2006) in his test found that a 61% improvement due to the use of AJAX. Case study done by (White, 2015) shows that AJAX application transferred on average 27% of the bytes that the traditional HTML application transferred. White (2015) reported a 73% performance increase for bytes transferred by using formula (HTML value – AJAX value) / (HTML value) * 100. Furthermore, the study also reported that, the AJAX application required fewer bytes to be transferred to carry out the tasks than did the HTML application and the users accomplished the assigned work in less time. Time savings result gained from the test indicate personnel cost savings. Google had used AJAX technology at its entire major product such as Google Map, Gmail, Orkut, Google Groups and Google Suggest. For instance, in Google Map, when user hold down the left mouse button and slide the cursor over an image on the AJAX-base Google Maps, part of the map not shown on the screen, the updates occur smoothly, and the image appears to move and change immediately (Paulson, 2005).

The new version of e-CEvas is applied with AJAX technology with the aims to increase the performance. Figure 3 shows the search interface that are used by the administrator of the system to search lecturer evaluation score. By searching the lecturer ID, the administrator can view all the related information of the lecturer. The administrator also could view the detail report of the lecturer including comments.



Staff ID : 188

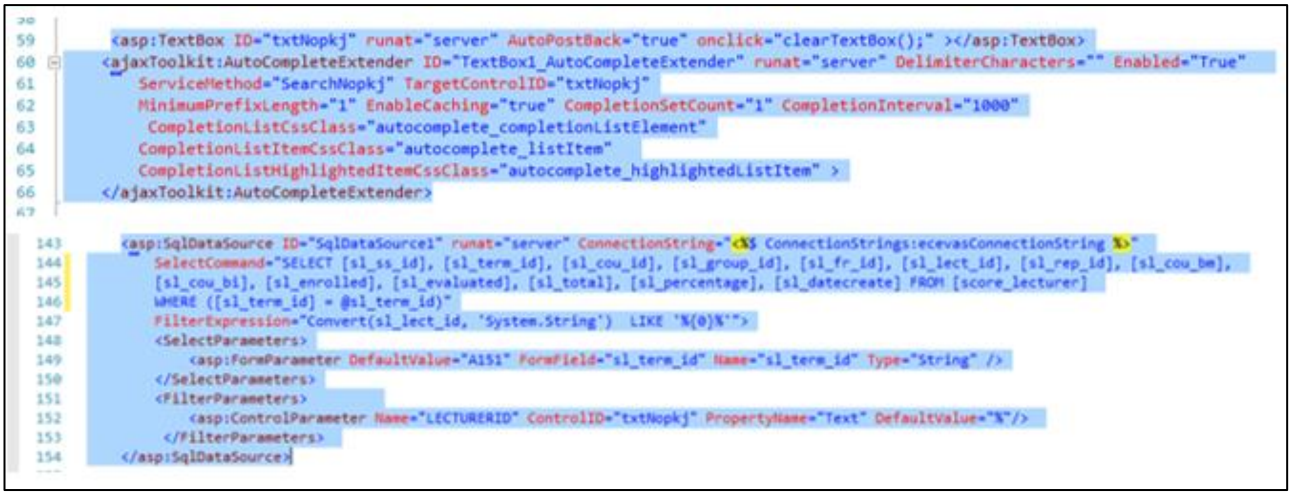
1880
1881
1885
1886
1887

Name : Dr. Muha[illegible]n Abu Bakar (1881)   Mode : Admin

NO.	SEMESTER	[illegible]	[illegible]P	COURSE	STUDENT ENROLLED	STUDENT EVALUATE	SCORE (%)	Action
2	A151	ST		ADVANCED DATABASE DESIGN	8	8	96.74	

Figure 3: Lecturer searching page with a list of lecturer score

Figure 4 shows code that are applied with AJAX used to create the searching data.



```
      <asp:TextBox ID="txtNopkj" runat="server" AutoPostBack="true" onclick="clearTextBox();" ></asp:TextBox>
    <ajaxToolkit:AutoCompleteExtender ID="TextBox1_AutoCompleteExtender" runat="server" DelimiterCharacters="" Enabled="True"
        ServiceMethod="SearchNopkj" TargetControlID="txtNopkj"
        MinimumPrefixLength="1" EnableCaching="true" CompletionSetCount="1" CompletionInterval="1000"
         CompletionListCssClass="autocomplete_completionListElement"
        CompletionListItemCssClass="autocomplete_listItem"
        CompletionListHighlightedItemCssClass="autocomplete_highlightedListItem" >
    </ajaxToolkit:AutoCompleteExtender>

    <asp:SqlDataSource ID="SqlDataSource1" runat="server" ConnectionString="<%$ ConnectionStrings:ecevasConnectionString %>"
        SelectCommand="SELECT [sl_ss_id], [sl_term_id], [sl_cou_id], [sl_group_id], [sl_fr_id], [sl_lect_id], [sl_rep_id], [sl_cou_bm],
        [sl_cou_bi], [sl_enrolled], [sl_evaluated], [sl_total], [sl_percentage], [sl_datecreate] FROM [score_lecturer]
        WHERE ([sl_term_id] = @sl_term_id)"
        FilterExpression="Convert(sl_lect_id, 'System.String')  LIKE '%{0}%'">
        <SelectParameters>
            <asp:FormParameter DefaultValue="A151" FormField="sl_term_id" Name="sl_term_id" Type="String" />
        </SelectParameters>
        <FilterParameters>
            <asp:ControlParameter Name="LECTURERID" ControlID="txtNopkj" PropertyName="Text" DefaultValue="%"/>
         </FilterParameters>
    </asp:SqlDataSource>
```

Figure 4: Searching page applied with AJAX code

AJAX ALTERNATIVES

AJAX can also be implemented with other technologies. For example, a simple Inline Frame could also be able to reload server - sided dynamic generated websites. But the disadvantage of iframes are security and compatibility issues. Iframes are neither dynamically producible. Other alternatives are LiveConnect functionality of Flash applications or JAVA implementations, but these technologies are manufacturer specific and it's difficult to implement extensions compared to AJAX. Another very new alternative is the DOM 3.0 load save specification which also allows several browser - server connections in the background. But the disadvantage is that it's not yet compatible in all browsers. AJAX combines the advantages of these alternatives without having their disadvantages.

WEB PERFORMANCE TESTING

Software product should have the software testing process to detect defects that exist when it runs. Conducting evaluation or testing process of a system or system component manually or automatically is necessary to verify that the programs developed meet user requirements. It could also be conducted to determine whether there are missing requirement results compared to actual results. In this study, a performance testing was executed to ensure the success of the system based on user requirements. Performance testing is used to check whether system meets its non-functional requirements identified in Software Requirement Specification (SRS) document (Kaushik & Fageria, 2014) or not. The testing also used to determine the response time and throughput of any web application (Sharmila & Rwasadevi, 2014). Performance testing tools are used to determine the time required to perform a task by the system. As for this study, the performance testing of the AJAX technology integrated in e-CEvas was tested using Microsoft Visual Studio web performance and load testing tool.

The main objective of the project testing is to test the performance of the ASP.NET system when integrate AJAX technology in some of its page. Some related factors that affect the performance of the application were identified. These factors are included as variables in the testing to enable a comparison of the performance. These factors are the hardware and the number of users accessing the system page which were in range of 1000, 2000 and 3000 lecturers and 2 administrators. The run duration for the testing is 10 minutes which sampling rate is every 10 seconds.

The key criteria for measuring speed of the applications is response time. "Response time" here refers to the time interval between client sending the request and receiving the response. Comparing the same application developed using two different method which were using ASP .NET coding integrate with AJAX and normal ASP.NET coding for determine the response time. ASP.NET pages used for testing are:
a) Normal ASP.NET page for administrator
i) http://localhost:81/Login.aspx
ii) http://localhost:81/Main/menu_pages.aspx
iii) http://localhost:81/Main/login_simple.aspx
iv) http://localhost:81/Lecturer/Lecture_view.aspx

b) ASP.NET integrate with AJAX technology for administrator
i) http://localhost/Login.aspx
ii) http://localhost/Main/menu_pages.aspx
iii) http://localhost/Main/Admin_ViewList.aspx

c) Normal ASP.NET pages for lecturer
i) http://localhost:81/Login.aspx

ii) http://localhost:81/Main/menu_pages.aspx
iii) http://localhost:81/Lecturer/Lecture_view.aspx

d) ASP.NET integrate with AJAX technology for lecturer
i) http://localhost/Login.aspx
ii) http://localhost/Main/menu_pages.aspx
iii) http://localhost/Lecturer/Lecture_View1.aspx

ASP.NET pages mentioned previously and part of application were developed in ASP.NET and AJAX. There are 4 tests conducted in this stage.

Test 1

In Test 1, four pages from normal ASP.NET pages for administrator requests and four pages from AJAX pages for administrator were generated from two users. Table 1 shows the response time results of three pages requests for normal ASP.NET and AJAX technology respectively. The response time is measured in seconds. The average response time for normal ASP.NET pages is 3.53s and for AJAX pages is 0.42s. Maximum response time of two requests for normal ASP.NET pages is 96.7s and for AJAX pages is 2.59s. Result shows that AJAX pages is faster than ASP.NET pages. Likewise, minimum response time is 0.30s for ASP.NET and 0.038s for AJAX pages. Total response time is 4.29s for AJAX and 96.33s for normal ASP.NET pages respectively.

Table 1: Response time for ASP.NET pages for administrator

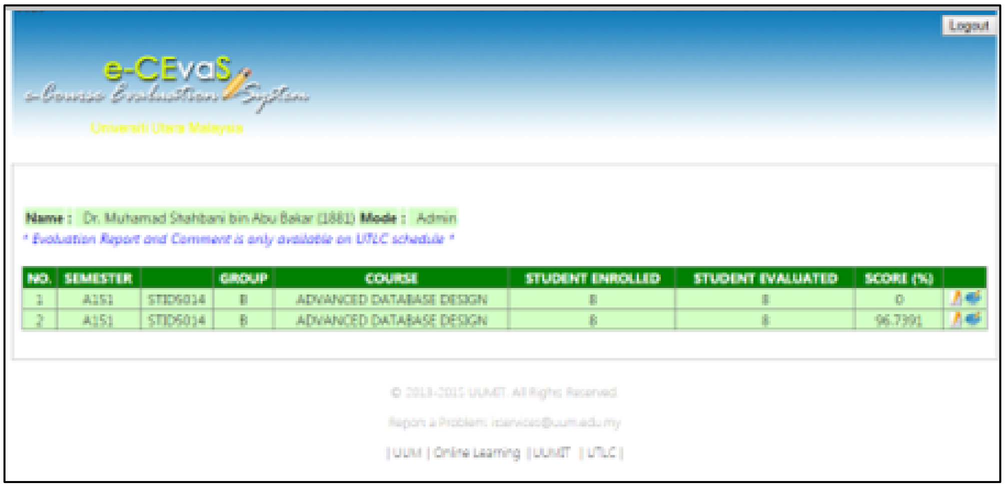
ASP.NET	Average response time (s)	Maximum time (s)	Minimum time (s)	Average test time (s)	Total response time (s)
Without AJAX	3.53	96.7	0.30	112	96.33
With AJAX	0.42	2.59	0.038	26.4	4.29

Test 2

In Test 2, the result is good as expected. Three pages from normal ASP.NET pages and AJAX pages for lecturer requests were generated from 1000 users. Table 2 shows the response time results of three pages requests for normal ASP.NET and AJAX respectively. The total response time for AJAX pages is 24.43s. However, total response time for normal ASP.NET pages is 34.89s. Figure 5 shows the interface that are used by lecturers view their evaluation score.

Table 2: Response time for ASP.NET pages for lecturer of 1000 users

ASP.NET	Average response time (s)	Maximum time (s)	Minimum time (s)	Average test time (s)	Total response time (s)
Without AJAX	12.3	29.3	1.83	55.1	34.89
With AJAX	8.06	16.4	1.02	32.3	24.43



Name : Dr. Muhamad Shahbani bin Abu Bakar (1881) **Mode :** Admin

** Evaluation Report and Comment is only available on UTLC schedule **

NO.	SEMESTER		GROUP	COURSE	STUDENT ENROLLED	STUDENT EVALUATED	SCORE (%)	
1	A151	STID5014	B	ADVANCED DATABASE DESIGN	8	8	0	
2	A151	STID5014	B	ADVANCED DATABASE DESIGN	8	8	96.7391	

© 2013-2015 UUMIT. All Rights Reserved

Report a Problem: icservices@uum.edu.my

| UUM | Online Learning | UUMIT | UTLC |

Figure 5: Lecturer view page

Test 3

In Test 3, the result still presents good performance of AJAX pages. The normal ASP.NET pages and AJAX pages for lecturer requests were generated from 2000 users. Table 3 shows the average test time for AJAX pages is faster than normal ASP.NET pages. The average response time for AJAX pages are 1.91 and normal ASP.NET pages are 2.38 which different in 0.47s.

Table 3: Response time for ASP.NET pages for lecturer of 2000 users

ASP.NET	Average response time (s)	Maximum time (s)	Minimum time (s)	Average test time (s)	Total response time (s)
Without AJAX	2.38	4.33	1.38	24.4	6.37
With AJAX	1.91	2.08	1.37	13.5	5.5

Test 4

In Test 4, the requests were generated from 3000 users simultaneously. Table 4 shows the response time results of three pages requests for normal ASP.NET and AJAX respectively. The result in this test shows poor performance from AJAX pages. Actually, AJAX pages depend on two other pages which uses code behind coding of server-side method. Many events and properties are associate with AJAX pages that has many coding for retrieving data from various table. Therefore, the server might take some time to run all the codes to display needed data.

Table 4: Response time for ASP.NET pages for lecturer of 3000 users

ASP.NET	Average response time (s)	Maximum time (s)	Minimum time (s)	Average test time (s)	Total response time (s)
Without AJAX	1.54	2.68	0.94	21.7	4.64

With AJAX	2.31	2.69	1.95	15.0	6.86

RESULTS AND FINDINGS

By analysing the results from the four tests that were conducted, it was found that variation of performance results depends on several factors. When the number of users is limited to two users, the response time was excellent as not much request to the system. The differences of response time are too obvious to show that AJAX is powerful web development techniques for programming. When the number of users increase, it had a great impact to performance of the system. Figure 6 show the total response time from different number of users.

With the increase number of users, the performance of the system was noticed to be decrease which indicate that there had been more loads on the application. This was cause by several other factors such as resources CPU, RAM and hard disk. There is a chance for performance degradation because of the low CPU processing and by using better CPU configuration it could handle a greater number of users. CPU processing is important factor that affecting the throughput. The CPU configuration must be rectified to improve the response time and as well as the throughput. This definitely will somehow prevent the application from crashing when number of users is more. System can perform much faster and gives excellent performance results when users increase with better processor speed to compare to the CPU used in this testing.

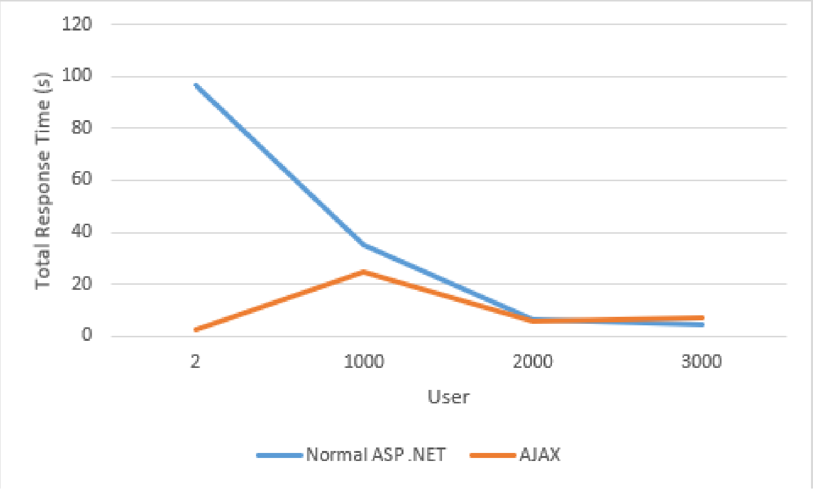


Figure 6: Summary of average response time based on number of users

Other factors that contribute to the response time and throughput result would be the implementing AJAX technology within them, the establishing connection to the database by which could be the higher impact on the average response time and throughput.

CONCLUSION AND RECOMMENDATION

It can be concluded that, the e-CEvas system was successfully designed and develop by using programming language stated by integrating AJAX technology in it. In Web Performance Testing, the purpose it was done is to determine the response time and throughput based on number of users used of the system. The results indicate that with increase number of users, it had a great impact to performance of the system. The performance of the system was noticed to be decrease which indicate that there had been more load on the application. This was cause by several other factors such as resources CPU, RAM and hard disk.

Based on the foregoing findings of the research, there are few recommendations that can be adapted in the system for future improvement of the system. Currently, not all module in the system uses AJAX technology in its programming. It would be better if all the module uses AJAX technology to fasten the information retrieval and other process. It is also making the server workload is minimized due to AJAX technology using only client-side programming compared to the normal use ASP.NET server-side programming. The data stored in the system database might be millions that could cause system crashing. By upgrading the CPU configuration, this will somehow prevent the application from crashing when number of users is more. The system would perform and function much faster and excellent performance results would be achieved when users increase with better processor speed to compare to the currently used CPU.

While we looked at the benefits of AJAX not only theoretically but also empirically in our study, we analyzed the drawbacks only in theory. In a future work it could be attempted to formalize and quantify the mentioned problems of the AJAX technology. Other future research could deal with a more comprehensive evaluation of other possible applications and include other usability aspects. This study show that the employment of AJAX brings a noticeable gain of efficiency and satisfaction in certain fields. Wisely used, Ajax can clearly improve the usability of web applications that way.

REFERENCES

Anderson, H., & J. Cain, et al. (2005). “Online Student Course Evaluation: Review of Literature and Pilot Study.” American Journal of Pharmaceutical Education; 69 (1) Article 5).

Kaushik, M., & Fageria, P. (2014). ”Performance testing tools: A comparative study”. International Journal of Innovative Science, Engineering & Technology.

Marchetto, A.T., & Ricca, P. F., (2008). State-Based Testing of AJAX web Applications. Software Testing, Verification and Validation. 1st International Conference, 2008 : pp. 121-130.

Merrill, C. (January 2006). "Using AJAX to Improve the Bandwidth Performance of Web Applications", Web Performance.com.

Murugesan, S. (2007). “Understanding Web 2.0.” IT Professional: 34-41.

Muzenda, A. (2013). Lecturers’Competences and Students’ Academic Performance. International Journal of Humanities and Social Science Invention WASSN. Volume 3 Issues 1. PP.06-13. Retrieve from http://www.ijhssi.org/papers/v3%281%29/Version 2/B3102 06013.pdf).

Paulson, L. (2005). “Building rich web applications with AJAX”, IEEE Computer, 38(10), 14-17.

Santiago, P., & Benavides, F. (December 2009). “Teacher Evaluation. A Conceptual Framework and examples of Country Practices”. 3-4.

Sharmila, S., & Ramadevi, E. (2014). ”Analysis of performance testing on web application”. International Journal of Advanced Research in Computer and Communication Engineering.

White, A. (May, 2015). “Measuring the Benefits of AJAX”. Retrieve from http://www.developer.com/xml/article.php/3554271.