I J C T A, 9(28) 2016, pp. 251-260

EMPIRICAL ANALYSIS OF SOFTWARE TESTING TOOLS USING EVALUATION CRITERIA

C.G. Anupama* Selva Kumara Samy Surya E*** and Sudarshan G******

Abstract: In today's world, software releases are being churned out faster than ever before. Customers are always particular that new releases must be implemented in a short time frame and without defects. This is where testing tools come in to play. The aim of this research is to do an empirical analysis of various testing tools, categorize the tools based on their features, and expound on possible improvements that will benefit the testing scene. This research followed a course of action that started with the identification of three testing tools to be evaluated. A suite of evaluation criteria was then developed, which was used to compare the tools. A target application was selected, which was then tested manually to find the existing defects. The survey questions were refined and were provided a weighted score based on the concepts of multivoting and decision matrix. A survey was then conducted to gauge the opinions of users about the various testing tools and how well they conformed to the previously decided evaluation criteria, using the selected application. The survey results were extracted in order to determine the level of performance of each tool. Finally, the most efficient testing tool was identified and inferences were made on how the other tools could be improved.

Keywords : Software testing, Test tools, Evaluation Criteria, Quality Criteria, Decision Matrix, Multivoting

1. INTRODUCTION

Testing plays a crucial role in software development because customers always have high expectations of quality from products, and the way to ensure that is by testing. The earlier the defects are identified, the easier and less expensive it is to correct them. Testing is the process of verifying that the product adheres to the requirements that were set for it. It is done with the intent of finding and fixing bugs, and making the software more reliable and of higher quality, thus ensuring customer satisfaction.

A plethora of testing tools exists today, and companies are not short of options when it comes to selecting appropriate tools to test their software. Even though many testing tools are available, acquiring a tool that has productivity and high quality gain is a difficult task. In order to find an efficient testing tool, organizations use various evaluation criteria to compare the existing tools. This evaluation process starts with identification of various user needs, followed by the establishment of an evaluation criteria, which is used to rate the selected tools [1] [2].

This research entailed the following steps as depicted in Fig.1:

1. A set of three tools were selected.
2. A suite of evaluation criterion was designed.
3. Identify the survey question scale.

* Assistant Professor, SRM University, **Email:** anupama.g@ktr.srmuniv.ac.in
** Assistant Professor, Department of Software Engineering SRM University **Email:** Selvakumarasamy.s@ktr.srmuniv.ac.in
*** Student, Department of Software Engineering, SRM University **Email:** Suryaelango85@gmail.com
**** Student, Department of Software Engineering, SRM University, **Email:** gsudarshanrajan@gmail.com

4. Refine and assign weightage score for questions.
5. A survey was conducted with sample sets of users.
6. The results of the survey were then extracted and were used in the determination of the best among the selected tools. Inferences from the experiment were then drawn and have been used in proposing possible improvements that can be made to the testing tools.

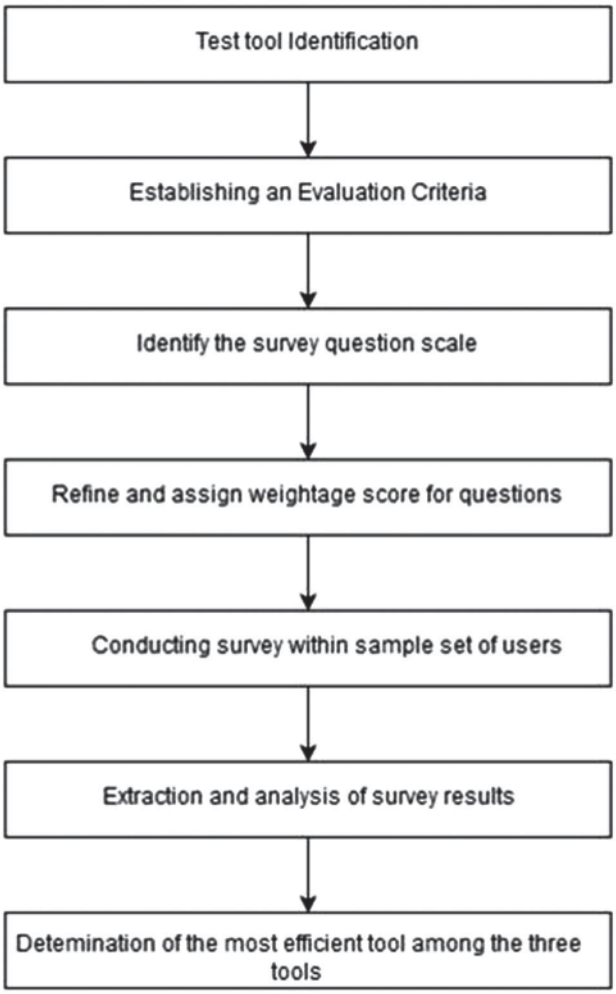

Figure 1. Test Tool Evaluation flowchart

2. TEST TOOL IDENTIFICATION

The three tools used in this research are a select few that are used in various organizations in order to make higher quality software that will result in customer satisfaction. These tools are compared using various evaluation criteria in order to find the most efficient tool according to a sample set of experts and users. The names of the testing tools are not mentioned and are only represented as Tool A, Tool B and Tool C.

2.1 Tool A

Tool A is an open-source, comprehensive tool used for automated testing. It has the capability to work on almost every Operating System. Being open source, it allows for cost effective testing, and it is easily

available to everyone. It supports a range of different programming languages like java, C#, ruby, python, and JavaScript. There is a uniformity among the different language bindings that applies to almost all the commands, so knowing any one language will suffice to translate the knowledge to other languages. It has a number of robust methods for locating elements such as CSS, DOM, etc. It is possible to perform simultaneous test using tool A, leveraging various browsers on various machines, which helps in cutting down costs for a large project. Tool A supports only web based applications and not windows based applications. Tool A doesn't provide a built in IDE for script generation, so it requires other IDE like eclipse for writing scripts. Report generation also requires external plugins.

2.2 Tool B

Tool B is a proprietary automation functional testing tool. It is an advanced and combined GUI and API testing tool, used to test the front-end and back-end of an application. It supports Business Process Testing, a component based testing framework that can be used to test the complete business process manually and automatically. It has a component reusability feature that makes for faster testing process. Tool B supports VB Script as a scripting language. It is the only language fully supported by the IDE. VB Script provides the advantages of Object Oriented programming, but not polymorphism or inheritance. Besides VB Script, tool B also partially supports java script and windows shell script. Tool B supports a wide variety of technologies like .Net, Oracle, Delphi, and so on. It supports all main browsers, and runs on OS-s like Windows (XP, Vista, 2003/7) and Windows Server (2008, 2012). It has an easily navigable IDE, and hence everything is provided in one place.

2.3 Tool C

Tool C is proprietary software which can be used to quickly validate the reliability and to provide an accurate view of performance of an application before it goes live. This testing software helps us to simulate and run workloads, benchmark system performance, diagnose problems and retest before launching a new or modified functionality. Tool C helps to detect bottlenecks in a new system and reduces hardware and software costs by accurately predicting application scalability and capacity. Tool C can emulate concurrent users to put the application through the rigors of real-life user loads, while collecting information from infrastructure components like Web servers, database servers etc. Tool C supports C Language, Visual Basic Scripting, Visual Basic for applications and Java Scripting. Scripts from other testing tools can be run by this tool by using an interface library. It supports Microsoft windows and Linux operating systems.

3. ESTABLISHING AN EVALUATION CRITERIA

The three test tools are compared based on evaluation criteria. Evaluation criteria are divided into quality criteria and functional criteria [3] [4].

3.1 Quality Criteria

Quality criteria are the benchmarks that need to be covered by the product in order to provide assurance to the customers that the product is of good quality. Quality criteria include functionality, usability, maintainability and efficiency, portability and reliability.

Different types of applications warrant different levels of importance to be given to each quality criteria. For example, a web application would require high priority to be given to usability and functionality, while embedded systems require more importance to be given to maintainability and efficiency. For the purpose of this survey, the target application chosen was a web application.

1. *Functionality:* Functionality refers to the product's characteristics like scalability, performance, interoperability, accurateness, security, and so on. With regard to the testing tool being surveyed, the aim is to find out how easy it is for users with different levels of experience to install the tool, how well the tool tabulates data about the target application, the technology support extended by the tool, whether it reveals the flaws in the target application's security mechanisms, whether the tool supports multiple browsers, whether it checks for unauthorized data access, whether it can distribute the test load to multiple computers, and so on.
2. *Usability*: Usability refers to how easy it is to use the tool. It involves understandability, learnability and operability. It probes whether the tool has an easily navigable interface, whether it has the ability to document test cases, whether easily understandable documentation is provided on how to use the tool, how easy or difficult it is to comprehend the error messages provided by the tool, etc. An ideal tool should be easy to use by both experienced and inexperienced users alike.
3. *Maintainability and Efficiency*: Maintainability and efficiency involve analyzability, changeability, stability and testability of the tool in question. The questions that arise with respect to this quality criterion are whether it is easy to make changes to the tool, how easy the tool is to maintain, whether the report contains a source file, whether the report generates the time taken to generate the tests, whether it is possible to add enhancements to the tool and so on[5].
4. *Portability*: Portability refers to how easy it is to use the tool on different operating systems and environments. It defines how easy it is to integrate with developer desktops, compilers, embedded target platforms, etc., and whether the tool can be exchanged between 32 bit and 64 bit systems, whether it is language independent , how difficult it is to install the tool on different devices, the ability of the tool to coexist with other tools, etc.
5. *Reliability*: Reliability encompasses maturity, fault tolerance and recoverability. It checks whether the tool measures defects consistently and reliably, whether it has test-retest stability, to what extent multiple instances can be run in a cooperative manner and share the load, whether the tool still works in the event of failure of some of its components, and how long it takes for the tool to recover if it crashes[6].

3.2 Functional Criteria

Functional criteria refer to the features that the tool possesses. It checks whether the tool does things like providing support for regression testing, whether it is possible to track the actual and estimated times for executing test cases, whether there is a feature that allows taking of snapshots, whether there is a feature for logging new bugs and checking the status of existing ones, whether it is possible to resume a suspended test case, the fault finding ability of the tool with respect to the target application, etc.

4. IDENTIFY THE SURVEY QUESTION SCALE

A survey is a quantitative research method for data collection where questions are used to describe a specific survey subject. In order to design an effective questionnaire, we used two types of survey question scales. They are dichotomous scale and likert-type scale respectively [7].

4.1 Dichotomous Scale

A question of dichotomous scale has only two possible responses. They are either yes or no.

4.2 Likert-type Scale

Likert-type scales are useful when we are measuring unobservable individual characteristics like attitude, feelings, opinions, etc. Likert-type use 5 or 7-point response scales most commonly. For the survey, we used 5-point response scale questions. The response scales use anchors such as 1 = Very high, 2 = High, 3 = Neutral, 4 = Low, 5 = Very low, etc.

5. REFINE AND ASSIGN WEIGHTED SCORE FOR SURVEY QUESTIONS

The survey questions are refined using the process of multivoting and are assigned weighted score by means of decision matrix [8].

5.1 Multivoting

Multivoting is used in the process of decision making to refine a list of items to just the high priority items. This process of narrowing down the options involves a brainstorming session within the group. First, all the options are listed out and numbered. Then the group sits together and reviews each item in the list. A method for voting is chosen. The vote is conducted and the results are tallied. Accordingly, the items in the list are reduced. In our research, this concept was used for the purpose of honing our list of survey questions down to 50, carefully selected and important questions, answers to which would help us gauge the extent to which certain quality criteria were covered by the tool in question. The functional criteria the questions aimed to target are: usability, functionality, reliability, portability, efficiency and maintainability, and functional criteria. A number of questions were gathered for each criterion. Multivoting was done on the questions to reduce the list to just the most relevant questions pertaining to each criterion. Each member of the group chose a certain number of, what was according to them, the most relevant questions. The redundant and low priority questions were thus weeded out.

5.2 Decision Matrix

A decision matrix is used in order to evaluate and prioritize a list of questions and calculate the weighted score for each question. . Initially a list of weighted criteria (functionality, usability, reliability, portability, efficiency and maintainability, functional criteria) is established and each question is then evaluated against those criteria. Decision matrix analysis is also known as Grid Analysis, Opportunity analysis, Pugh Matrix Analysis, criteria-based matrix analysis, Multi-Attribute Utility Theory etc. The entire procedure of decision matrix analysis is divided into 4 steps [11].

Procedure

Step 1: All questions are listed as the row labels on the table, and criteria are listed as the column headings.

Step 2: Scoring each question against each criterion based upon the relative importance that each question has for that criteria down the column from 1 (poor) to 5 (very good).

Step 3: The next step is working out the relative importance of the criteria. The importance is scored from 1 to 5, where 1 means that the criterion is absolutely unimportant to the testing tool, and 5 means that it is very important.

Step 4: Multiply each question's rating by the weight of the criteria. Add the points for each question and the final sum of a question for all the criteria is considered as its weighted score.

$$\text{Weighted score} = \sum (\text{rating of a question with respect to a criterion} * \text{weight of the criterion})$$

After brainstorming with experts, the weighed score of a question was decided and is represented as sum in Table 1.

Table 1.
Quality and Functional Criteria

Q. No.	*Functio-nality*	*Usab ility*	*Maintain ability & efficiency*	*Porta bility*	*Relia bility*	*Func tional Criteria*	*Sum*
1	5	4	2	3	3	3	80
2	5	4	3	3	4	4	90
3	5	2	4	2	2	2	67
4	5	3	4	2	3	3	78
5	5	1	4	1	2	1	56
6	5	3	3	1	3	4	74
7	5	4	3	1	3	4	78
8	5	4	3	1	4	2	76
9	2	5	2	2	3	2	62
10	4	5	2	2	4	3	79
11	4	5	3	2	3	4	81
12	2	5	3	4	3	2	73
13	2	5	4	1	2	4	66
14	1	5	1	1	3	4	56
15	1	5	2	1	2	4	55
16	3	2	5	2	4	3	71
17	2	4	5	1	4	3	70
18	3	4	5	1	4	4	78
19	3	1	5	2	3	1	57
20	4	3	5	2	1	1	62
21	3	4	5	1	4	4	78
22	2	2	5	3	1	4	61
23	3	2	4	5	2	4	75
24	2	4	2	5	3	1	67
25	3	2	4	5	2	4	75
26	4	2	2	5	2	2	68
27	2	4	2	5	3	2	70
28	3	3	2	5	4	2	75
29	4	3	3	5	3	2	79
30	4	2	3	5	2	2	71
31	4	3	3	5	3	2	79
32	2	3	3	5	4	2	73
33	3	2	4	2	5	3	72
34	3	3	2	1	5	4	69
35	2	2	2	2	5	4	64
36	2	1	4	3	5	4	70
37	2	2	3	1	5	4	63
38	4	1	4	2	5	2	70
39	4	2	3	1	5	3	70
40	2	2	4	3	3	5	69

Table 1 Contd...

41	4	3	2	1	3	5	69
42	4	2	3	3	2	5	72
43	2	3	3	1	4	5	66
44	2	3	3	1	4	5	66
45	4	1	3	1	4	5	68
46	1	3	1	1	3	5	51
47	3	3	3	1	4	5	71
48	2	4	2	1	3	5	63
49	1	4	1	1	4	5	59
50	2	4	3	1	3	5	66

6. CONDUCTING SURVEY WITH SAMPLE SET OF USERS

For the purpose of this research, a survey was conducted with a sample set of users with different levels of expertise and experience ranging from first time users to experts. The testers were given a set of 50 questions .The questions were of 2 types.25 questions were of dichotomous scale and the rest were of likert-type scale.

Calculation : The score given by the user for a question is calculated as given in table 2.

Table 2.
Score Calculation

Score Given by user	*Response given by user*	*Type of question*
1	Yes	Dichotomous Scale
0	No	
1	Very high	Likert-type scale
2	High	
3	Neutral	
4	Low	
5	Very low	

Let the total number of users attended the survey be *n*.

Then the total score of a question is calculated as,

$$\sum_{k=0}^{n} \text{weighted score of a question*score given by user } k$$

The maximum score of a question is calculated as,

(a) If the question is of dichotomous scale:

$$\sum_{k=0}^{n} \text{weighted score of a question} * 1$$

(b) If the question is of likert-type scale:

$$\sum_{k=0}^{n} \text{weighted score of a question} * 5$$

The total score of a criterion c consisting of m questions,

$$\sum_{k=0}^{m} \text{Total score of question satisfying } c$$

The maximum score of a criterion c consisting of m questions,

$\sum_{k=0}^{m}$ Maximum score of question satisfying c

$$\text{Percentage of criterion satisfied by a tool} = \frac{\text{Total score of criterion}}{\text{Maximum score of criterion}} * 100$$

After the survey results were obtained from users, the percentage of criterion satisfied by a tool was calculated using the above formulas.

The percentage of criterion satisfied by Tool A, Tool B and Tool C is depicted in Figure.2

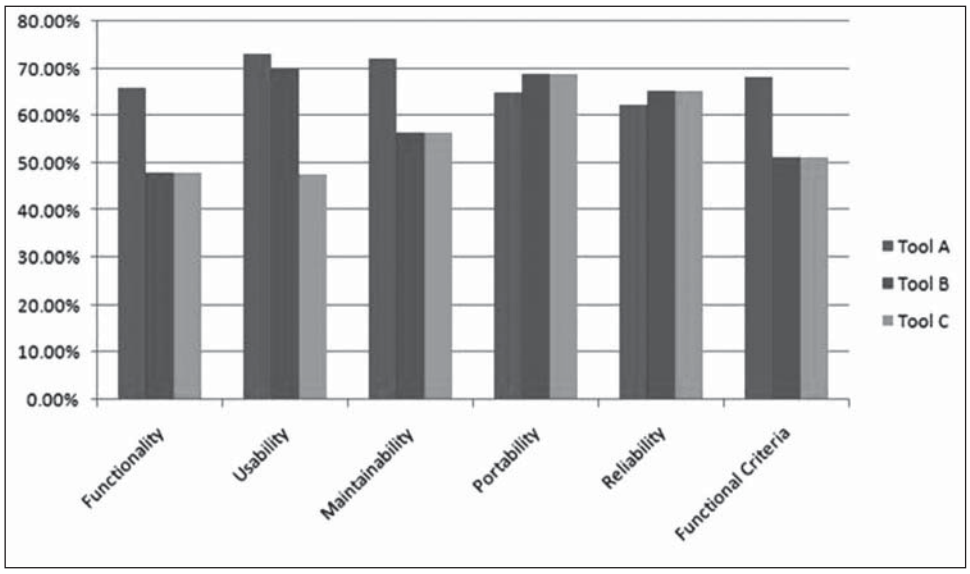

Figure 2. Percentage of Criterion satisfied by Tool A, Tool B and Tool C

7. DETERMINATION OF THE MOST EFFICIENT TOOL

Based on our study, the following results were obtained.

(a) In terms of functionality, Tool A is more efficient compared to the other two tools. The percentage of functionality satisfied by Tool B and Tool C are almost same according to our study.

(b) In terms of usability, Tool A is more efficient. Tool B is a little behind Tool A in terms of this criterion.

(c) In terms of maintainability and efficiency criterion, Tool A is more efficient compared to the other two tools. The percentage of maintainability and efficiency satisfied by Tool B and Tool C are almost same according to our study.

(d) In terms of portability, Tool B and Tool C are more efficient compared to Tool A.

(e) In terms of reliability, Tool B and Tool C are more efficient compared to Tool A.

(f) In terms of functional criterion, Tool A is more efficient compared to the other two tools. The percentage of functional criteria satisfied by Tool B and Tool C are almost same according to our study.

Using the extracted results, we identified the most efficient tool out of the three tools as depicted in Figure.3.

According to our study, Tool A is the most efficient testing tool out of the three selected tools.

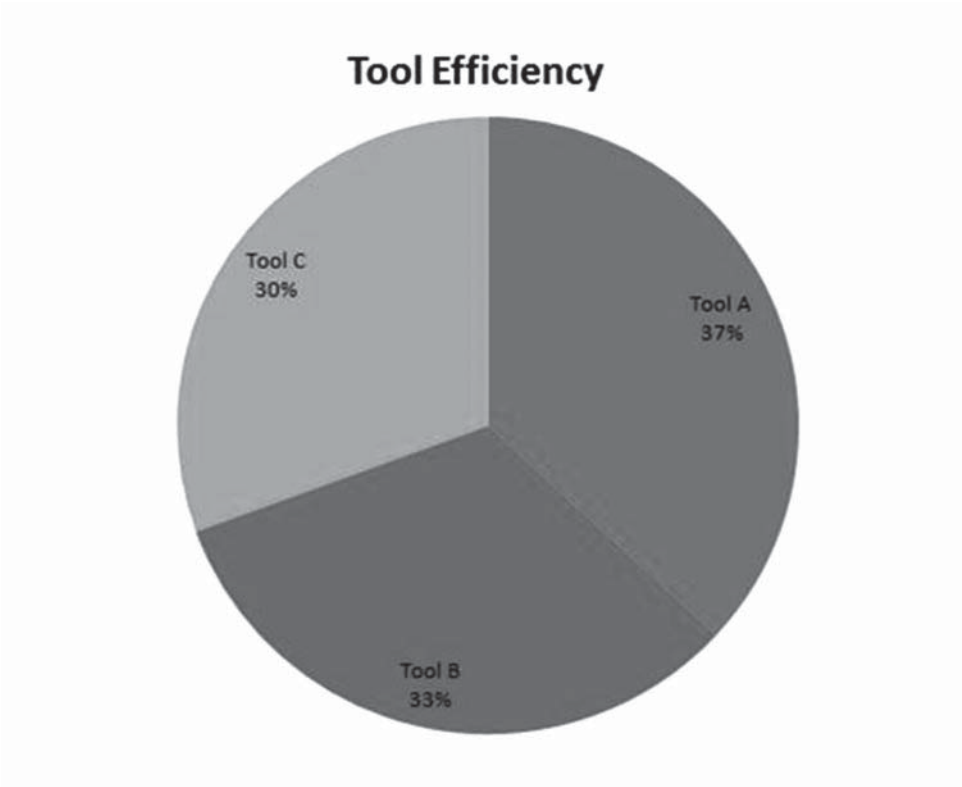

Figure 3. Efficiency of Tool A, Tool B and Tool C.

8. FUTURE ENHANCEMENTS

Based on the concept of multivoting and decision matrix, an evaluation tool can be designed that can make it easier for companies to compare various testing tools and to select an ideal testing tool among the selected ones.

9. CONCLUSION

In this paper, three testing tools were evaluated against a set of quality criteria, and compared with each other. A target web application was used by the users of the survey, to test all three tools. Tool A was found to be the top runner for functionality, Tool A for usability, Tool A for maintainability and efficiency, Tool B and Tool C for portability, Tool B and Tool C for reliability and tool A for functional criteria. The concepts used for deriving the results were 'multivoting' and 'decision matrix'. Overall, tool A was found to perform the best.

References

1. Michael P Sexton Robert M Poston, "Evaluating and Selecting Testing Tools",IEEE Software, vol. 9, no. 3, pp. 33-42, May 1992.

2. Nancy Bordelon, "A Comparison of Automated Software Testing Tools "Annals of the MS in Computer Science and Information Systems at UNC Wilmington, Vol.6 N1 Paper 1,2012, http://csbapp.uncw.edu/data/MSCSIS/annalspaper.aspx?v=6&i=1&p=1(last visited: April 2016)

3. T.Illes, A.Herrmann, B. Paech, J. Rückert,"Criteria for Software Testing Tool Evaluation – A Task Oriented View", Proceedings of the 3rd World Congress for Software Quality 2005, Vol.2 .pp.213-222.

4. Tanja E.J. Vos, Beatriz Mar´ın, Maria Jose Escalona, Alessandro Marchetto," A Methodological Framework for Evaluating Software Testing Techniques and Tools,12th International Congress on Software Quality,2012

5. What is Maintainability testing in software?, http://istqbexamcertification.com/what-is-maintainability-testing-in-software/, (Last visited: April 2016)

6. Colin Phelan, Julie Wren, Exploring Reliability in Academic Assessment, https://www.uni.edu/chfasoa/reliabilityandvalidity.htm, (Last Visited: April 2016)

7. How to Create a Survey–Survey Question Types with Examples, http://www.survio.com/en/blog/tips-and-tricks/how-to-create-a-survey-survey-question-types#.VxHnSXrmzIU (Last Visited: April2016)

8. Nancy R. Tague, "*The Quality Toolbox", Second Edition,* ASQ Quality Press, 2004, pages 219–223.

9. Master Thesis Title: Evaluation of Software Testing Tools-87710, http://myassignmenthelp.info/assignments/master-thesis-title-evaluation-software-testing-tools-87710/, (Last visited: April 2016)

10. **Checklist for Testing the Compatibility and Portability of Web Applications,** http://www.softwaretestinggenius.com/checklist-for-testing-the-compatibility-and-portability-of-web-applications, (Last Visited: April 2016)

11. Decision Matrix Analysis-Making a Decision by Weighing up Different Factors, https://www.mindtools.com/pages/article/newTED_03.htm, (Last visited: April2016)