Remote Evaluation Of An Interface Usability Using Asynchronous Auto-Logging Methodology

Tarus Nicholas Kiprotich, Stephen Kimani, Wilson Cheruiyot

Abstract: The need of developing more usable web applications has motivated the development of a number of techniques, methods and tools to address web usability issues. In recent years many automated remote usability evaluation methods have been employed. Despite the advantages offered by these methods, usability evaluation is still a difficult and a time consuming task partly due to complexity of the tools used and partly due to expertise required. Analysing data obtained from web server logs can reveal usage pattern of users. This gives a highly improved understanding of users' behaviour. This information can then be utilized in improving a web e application way which usability interfa evaluation of a web application can be approached. State transition network (STN) was preferred in this work. This paper presents a model for automated remote usability evaluation which automatically generates specific usability information of a remote web application.

Index Terms: asynchronous, auto-logging, click-stream, dialogue design, Paths, usability, violations

1 INTRODUCTION

Since early years of existence, internet has made massive advances in transformation into a form we know of today [1]. Web applications have come to play a central role in the world of modern business and information dissemination. This shift has in turn imposed a need of professionals to develop and maintain web applications. Developing and maintaining web applications is not enough. It is the level of usability of this web applications that determines effectiveness and efficiency to achieve a desired goal. Usability of an interactive software application is seated within the design of the user interaction components [2] and therefore, it is crucial to do usability evaluation of a web application interface. Evaluation of users in their real work environment is very costly, this is an obstacle to sound evaluation. Increased internet connection in working environments makes it possible to collect users' session data and automating analysis is possible for identification of usability issues. Automated and traditional usability evaluation are equally efficient and effective for application usability evaluation [3] therefore, automated remote evaluation is a promising area to pursue. There has been extensive work relating to automating usability evaluation which has resulted to development of a number of tools which perform website usability analysis. One of the reasons for automating usability evaluation is to reduce the cost of hiring an expert. Various academic papers and prototype have been developed in order to try and automate web application usability evaluation remotely. These academic papers and prototype have had different degrees of success. Automation can provide excellent solution towards overcoming limitation of having expertise in usability evaluation and hence, developers will concentrate in developing and improving existing applications using information from remote tools. Despite advantages that automatic usability evaluation provides, it has been observed that with advances in technology, automation has never been effective enough to enable tools to generate specific usability information and even further critique a web application. It has also been noted that there is inconsistency in results generated as each tool gives different information for same application interface. Past attempts of automation have also resulted in tools that are too specific and application related, very costly and oversimplified [4]. Tools for evaluation should be in conjunction with standard usability evaluation techniques [5] and information should also be presented in a format that is easily understood by developers. This paper presents asynchronous auto-logging methodology model that follows pre-defined clickstream pattern which are derived from dialogue designs (STN). Logging-on the user clickstream in order to capture user generated events is done. Analysis of this clickstreams can provide useful hints regarding possible usability problem. This model allows developers to directly get specific information regarding usability of their web applications. Ivory and Hearst [6] provide a good discussion of tools for usability evaluation according to a taxonomy based on four dimensions: method class (the type of evaluation), method type (how the evaluation is conducted), automation type (the evaluation aspect that is automated) and effort level (the type of method required to execute the method). According to this classification, Asynchronous auto-logging solution for usability testing involves: automatic capturing logs generated at client-side and supporting automatic analysis to ease the identification of the usability issues. Google Analytics [7] has the potential to be configured to capture custom events at client-side and it offers a number of statistics information and reports, but it is rather limited in terms of number of events that it captures for each session [8]. Model-based approaches have been used to support usability evaluation. An example is WebRemUsine [9] which is a tool for remote usability evaluation of web applications through browser logs and task models. This tool require expertise to use and it is also time consuming to a developer. WUP[1] is a tool that captures logs generated at client side, the tool will then support in automatic analysis of the captured logs. Users perform some predefined tasks specified by an evaluator.

WUP- web usability probe

Evaluation is carried out by comparing the click-stream patterns of a user and a developer. A difference in steps can imply a potential usability problem [8]. WUP is a good tool for usability evaluation as it is easy to use but, users must be willing to participate during evaluation session as an instructor provides instructions using a dialogue box. OLAP is an approach for interactively exploring multi-dimensional data. Relationship among various dimensions are explored in order to make inference about the data. OLAP may enable the evaluators to identify areas of the site that need to be improved but interactive exploration and

summarisation of web log data that is possible with OLAP is in stark contrast to statistical analysis approaches [10]. Automated inquiry are used to measure user satisfaction. Users are asked to fill a questionnaire or make a comment on a dialogue box. The data collected will be analysed to generate information that is used to describe usability of a web application. This describes the overall usability of a web interface. SUMI[3] is one of the tools used in automated inquiry. It provides results based on extensive standardisation database e.g. spreadsheet [11]. The results are very reliable since database allows evaluation of data against what is considered to a standard usability issues. SUMI results are analysed by a computer programme called SUMISCO. This is the simplest approach but it requires the efforts of web application user to get the data. The feedback from the user could be subjective. It is also time consuming and determination of specific usability issue requires expertise.

1 Asynchronous Auto-Logging Methodology.

2.1 Dialogue Designs.

One important aspect of HCI is the dialogue which is the interaction that takes place between a human user and the computer. A dialogue refers to the structure of the interaction. Dialogue design helps to understand a web application design better. Formal representation makes it possible to analyse dialogues to identify usability problems. There are several formalism that can be used to represent dialogues [12]. STN was preferred in this work because they are the most intuitive among all formalisms. It assumes that a dialogue essentially refers to a progression from one state of the system to the next in the system state space.

2.2 Determination of Paths.

The model supports evaluation of web application by exploiting clickstream pattern and comparing them with pre-defined paths. It is based on an intermediate proxy server whose purpose is to capture log file from remote users. The model does not require use of plug-in installation in the client side or specific client configuration (figure 1)

[2]OLAP- On-line Analytical processing
[3]SUMI- Software usability measurement inventory

The script used by the model are stored in the proxy server and thus there is no security conflict when the page is accessed from the client since the page appears as coming from the proxy server. As users carry out their tasks, clickstreams are recorded and accessed by the model for analysis of specific usability issues. A link from the start to the end form a path. A path has a specific goal. A menu in a website makes the start of a path and the last link is the end of a path. Each path has a specific goal as illustrated in table 1 A clickstream pattern that does not follow the predefined click paths are considered as a violation. Paths created using dialogue design eases identification of usability issues, hence developers will access the information from remote tools. All logged data are sent asynchronously to the model while the users move from one page to another.

2.3 Determination of usability issues using paths.

The development of a proxy-based model considering client-side data encounters different challenges regarding to identification of specific usability issues from user interaction data. For example a user may move from link 2 to 4 then back to 2 and then 5 (Figure 2). The intention of the user could be to compare features/cost of different cameras and then select one. The pattern may imply a usability issue according to the models' design. The solutions that were adopted includes time for a particular task was assigning a minimum time to a particular task and creating associations among.

2.4 Testing of Asynchronous Auto-Logging Methodology.

Thirty-five participants were recruited for the purpose of this study from Visions Interactive and InterPay. All of them are employees of the IT companies based in Nairobi. Research was done in the two companies separately. The test was divided into two parts. To save on company's time, the second task was divided into six parts. This is as a result of the six approaches of automated remote usability under study. One tool from each of these approaches was selected randomly depending on its availability. Jumia website was chosen for this study. The participants were screened for their knowledge in IT and familiarity in usability evaluation using a questionnaire. A test using the website was conducted using a list of pre-defined task. These tasks were a representative of scenarios that a regular user would experience when using a website. These tasks provided an opportunity to gather information on variables such as time spent on completing a task. Participants were then provided with a list of tasks with detailed instruction to perform. www.jumia.co.ke_is a website for Jumia Company used in this study. The company trades in phones laptops, cameras and their accessories. Log files in each browser were taken for use in the determination of potential usability issues. From participants screening, eleven of them were identified to be having sufficient IT skills and two of them were conversant with usability evaluation methods. Therefore, one was assigned the task of being a trainer and further trained on the use of three of the selected tools. Six groups of two participants were made. The six groups were further divided into two groups for training on the use of the model and the selected tools (Table 2). One was trained by the researcher and the other group by the identified trainer. The researcher provided assistance only if the participants were unable to complete the task and requested for assistance. At the end of the performance test, the evaluator collected the subjective feedback and then debriefed the participants and gave thanks to them.

3. Tables and Figures

Figure 1 *Asynchronous remote usability model*

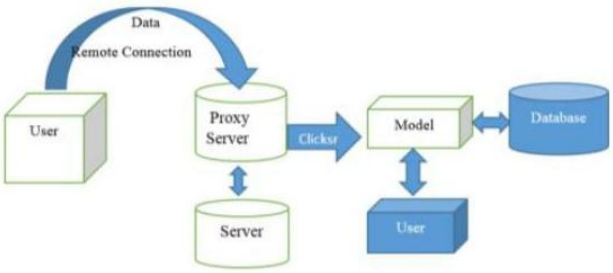

Figure 2 *Links used in creation of paths*

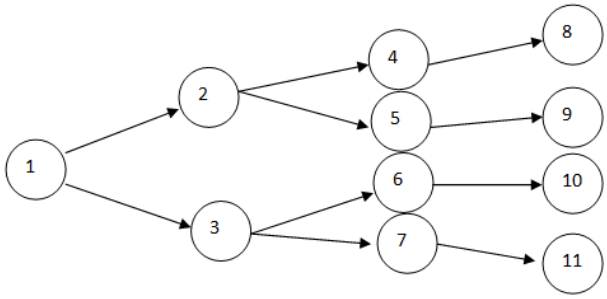

Table 1 *Information presentation to developers*

Path	Goal	Use rs	Violations	Significance	Cause
1	Price/Features-Samsung camera				
2	Price/Features-Sony Laptop				

Table 2 *Groupings used in research*

	Trainer 1			Trainer 2		
Tools	Model	WUP	WRU	OLAP	GA	SUMI
S 1P	1,2	3,4	5,6	7,8	9,10	11,12
S 2P	11,12	10,9	8,7	6,5	4,3	2,1

S 1-Session 1 S 2- Session 2 GA-google Analytic
WRU-WebRemUsine P- Participants

This approach satisfies a number of requirements about evaluation tools. It works in different configurations of hardware and software, does not depend on specific configurations, it does not impact on the web site usage and it does not interfere with the web page. It supports web application interface evaluation of geographically distributed, diverse users who are in their native work environments. Given that the model automatically generates usability test results, it therefore appears to be a viable alternative to the other automated remote usability evaluation approaches. The model provides dynamically an indication of the number of sessions that have been logged by users. Association among links and tasks provides an excellent approach for determination of violations. This created a dependency in which a task is performed before the current task. For each path there is an indication of the goal, number of users, violations and significance. These features allows remote users to freely perform tasks and the developers get information directly which is automatically generated. This information will be used to improve the existing web applications. The starting URL provides flexibility in this evaluation approach since it allows the model to define that certain tasks have to start first in different menus of the evaluated web site. Finally, the dependency feature among tasks is provided in order to make possible to define evaluations where one tasks is mandatory (e.g., login) in order to perform others (e.g., creation of content in a login protected Web site). Once some users have actually performed a task, a developer can access information in tabular and graphical formats. This approach provides usability information in tabular and graphical representations. These representations are easily understood by developers. The model can also be interactively set in order to identify more information. Violations are lined up from the highest to the lowest and this allows the model to easily determine usability issues from those violations whose significance is considered to be high.

4. Conclusion

This work proposed a new methodology for conducting an automated remote web application usability and empirically compared it with selected tools from the automated approaches of remote usability evaluation. One of the most important findings of this study is that this approach is more effective than other automated approaches in terms of the time taken to identify usability issues and determination of specific usability problem. Interestingly, participants recruited to carry out a test of the model and some selected tools from other approaches appeared to identify a slightly larger number of usability issues than expected by the researcher. Test participants experienced lower overall workload in the developed model than in either of the automated remote testing tools. The findings indicate that the test participants experienced a higher level of involvement in the other approaches' tools than in the model. Asynchronous auto-logging model offers a remote testing infrastructure without any software installation required by the usability test participants and facilitators. The subjective ratings provided by the test participants in the final subjective rating questionnaire revealed significant differences in terms of user satisfaction and ease of use. The user satisfaction and ease of use were higher for the model and the automated inquiry than for the other environments. However, some of the test participants commented that they felt some pressure to avoid making mistakes while being observed by a test facilitator. Test participants felt that they were best able to analyse the user interaction with the web interface in the models' environment. They felt they were least able to analyse the user interaction in the metric and task based environment. The low rating of these environment on metrics was probably the result of a technicality of tools used in these

environments. Although there was no significant difference between the evaluation approaches in the total number of usability issues identified, there was a significant difference between the methods when it came to the amount of time used to identify this usability issues. Another interesting piece of information seemed to emerge when the subjective feedback on future preference was analysed. Results showed that participants preferred to use the model for usability evaluation in future. The analysis of subjective ratings feedback on five variables (comfort, ease, realness, convenience, and future) revealed that although the results of the data analysis on overall rating of evaluation methods showed significant differences, the data on further scrutiny revealed that the range of the rating scale might not have been sensitive enough to collect user feedback on their overall experiences in participation. The model retains the inherent advantages of asynchronous usability testing, including significant savings in travel time and cost and evaluation of geographically dispersed users.

References

[1] http://www.hindawi.com/journals/ahci/2014/479286/ accessed on 15/08/2015

[2] Cray Woolsey (2013). Usability methods and tools. http://research.cs.vt.edu/usability/index.html.

[3] Betra, S & Bishu, R (2007), Web usability evaluation: issues and concerns usability and internalization, HCI and culture, LNCS 4559, pp 243-249.

[4] J. Mifsud, USEFul (2011),: a framework to mainstream web site usability through automated evaluation [B.Sc. thesis], University of London.

[5] Justin Misfud (2011),: Mainstreaming usability through automated usability evaluation, usabilitygeek.com/mainstreaming-web-site-usability-through-automated Accessed on 09/09/2015

[6] Ivory, M.Y., Hearst, M.A (2001).: The state of the art in automating usability evaluation of user interfaces. ACM Comput. Surv. 33(4), 470–516

[7] Google: Google Analytics (2009), :http://www.google.com/analytics/index.html

[8] Tonio C, Fabio P, and Vagner F (2011),: Web Usability Probe: A Tool for Supporting Remote Usability Evaluation of Web Sites. CNR-ISTI, HIIS Laboratory, Via Moruzzi 1, 56124 Pisa, Italy

[9] Paganelli, L., Paterno, F(2003),: Tools for Remote Usability Evaluation of Web Applications through Browser Logs and Task Models. Behaviour Research Methods, Instruments, and Computers 35(3), 369–378

[10] https://books.google.com/books?isbn=9401703752 accesd in 0/08/15

[11] Kadamula V and Silvia P (2012): Human Comfort and Security of Information Systems: Advanced Interface for the information society springs Science and Business media.

[12] Pradeep, Y and Sammit B: Human Computer Interactions- An IntroductionNPTEL.AC.IN/COURSES/106103115/MODULES7/1.PDF ACCESSED ON 02/09/2015

[13] First Author Tarus Nicholas Kiprotich is a holder of B.Ed. (Sci) Mathematics and Physics from KU (2005), Ongoing MSc. Computer System Jomo Kenyatta University of Agriculture and Technology (JKUAT). Worked at ECK (Electoral commission of Kenya), Independent Election and Boundaries Commission (IEBC) and currently working at Teachers Service Commission (TSC) Nandi North as a teacher of mathematics and physics.

[14] Second Author Dr Stephen Kimani: He holds Certificate in Commercialisation from University of Tasmania, Australia. Received PhD in Computer Engineering DIAG, Sapienza University of Rome, Italy. MSc in Advanced Computing (HCI & Multimedia) University of Bristol, UK. BSc First Class Honours (Mathematics & Computer Science) Jomo Kenyatta University of Agriculture and Technology (JKUAT), Kenya Director School of Computing and Information Technology, JKUAT Nairobi, Kenya. His research interest include but not limited to: Human-Computer Interaction (HCI) as it relates to areas such as: User Interfaces, Usability, Accessibility, Evaluation, Mobile Computing, and Information Visualization.

[15] Third Author Dr.Cheruiyot W. Kipruto: Received B.sc (Mathematics & Computer Science from Jomo Kenyatta University of Agriculture & Technology (JKUAT), Kenya; Msc. in Computer Application Technology, Central South University(CSU), P.R.China; PhD in Computer Science &Technology ,Central South University(CSU) Changsha P.R.China. Serving as a Deputy Director (Postgraduate Programmes) and Senior lecturer in the School of Computing & Information Technology at JKUAT, Kenya. His Research interests include but not limited to: Multimedia Data Retrieval, Internet of Things, Evolutionary Computation for Optimization, Digital Image Processing and ICT for Development.