Improving Quality in Use of Web Applications in a Systematic Way

Philip Lew[1] and Luis Olsina[2]

[1] School of Software, Beihang University, China
[2] GIDIS&Web, Engineering School, Universidad Nacional de La Pampa, Argentina
philiplew@gmail.com, olsinal@ing.unlpam.edu.ar

Abstract. A first step to evaluate quality is to define nonfunctional requirements usually through quality models. The ISO 25010 standard describes one such model for general usage in specifying and evaluating software quality requirements, but its concepts need to be adapted based on a specific information need and context, i.e. for evaluating WebApps in a real situation particularly when it comes to evaluating quality in use (QinU). WebApps and their quality evaluation have been proposed in research through many approaches, but mostly for the purpose of understanding, rather than improving. In this tutorial, we demonstrate employing a quality modeling framework and strategy to instantiate quality models with the specific purpose not only to understand the current situation of a WebApp, but also to improve it.

Keywords: Quality in use, Actual usability, Improvement, SIQinU strategy.

1 Tutorial Contents

Web applications (WebApps) have evolved considerably since the simple informational and ecommerce websites of the 90's. We start by giving an overview of the Web eras as well as the unique intrinsic features of WebApps. This characterization will help us identifying/mapping quality features. As background, the ISO 25010 [1] characteristics and the relationship among internal quality, external quality, and QinU models are discussed. We argue the need to model *information quality* and *learnability in use* dimensions for measuring and evaluating quality for WebApps which is absent in this standard.

Then, we illustrate how to instantiate quality models for the purpose of improvement assuming that understanding is the means and improvement is the ultimate goal. With evaluation as a step toward improvement, quality models must be instantiated with this in mind. So, by using the 2Q2U (*Quality, Quality in use, Usability and User experience*) modeling framework [2, 4], we show how to instantiate external quality (EQ) and QinU models with the goal of improving. These models are developed and used while executing a specific strategy for improving QinU for WebApps, namely SIQinU (*Strategy for Improving Quality in Use*) [3].

SIQinU has a six-phased process which starts with identifying problems in the QinU of a WebApp and then characterizing these problems using an instantiated QinU model. These problems come from evaluating a WebApp in a real context of

A. Harth and N. Koch (Eds.): ICWE 2011 Workshops, LNCS 7059, pp. 359–360, 2011.

use, collecting data directly from the application through the use of weblogs. Weblogs are not new, but this approach is novel in that our objective in data collection is not related to where users go and how long they stay on each page, but rather related to when executing planned real tasks. These problems in QinU are then mapped to potential problems in the WebApp's EQ, or intrinsic properties which lead to deriving a specific instantiated EQ model for the WebApp that is tied to the problems in QinU. This is followed by evaluating the WebApp using this EQ model resulting in an EQ benchmark. The EQ benchmark gives us indication of poorly performing EQ attributes and these poorly performing attributes give us the basis to make recommendations for improvements to the WebApp. After recommendations have been implemented, we evaluate EQ again to determine where and to what degree EQ was improved. Then finally, we use the same context of use as in the first phase, using real users executing a real task, and evaluate QinU again to determine the improvement gained as a result of the changes in EQ and improvements just made.

Ultimately, in the process of using SIQinU, we are able to gain insight regarding the *depends on* and *influences* relationships [1] for the particular 2Q2U instantiated models, and their characteristics and attributes driven by our purpose to improve. In addition, we can continue to iterate the SIQinU improvement cycle to gain further insight and granularity adding a temporal component for later study. Finally, in the tutorial, we illustrate SIQinU using JIRA (www.atlassian.com), a well-known defect tracking WebApp, by specifying a task designed to collect information at the sub-task level so that specific screens and their properties (EQ attributes) could be identified for potential problems leading to poor performance in QinU.

References

1. ISO/IEC 25010:2011(E): Systems and software engineering. Systems and software Quality Requirements and Evaluation (SQuaRE). System and software quality models (2011)
2. Lew, P., Olsina, L., Zhang, L.: Quality, Quality in Use, Actual Usability and User Experience as Key Drivers for Web Application Evaluation. In: Benatallah, B., Casati, F., Kappel, G., Rossi, G. (eds.) ICWE 2010. LNCS, vol. 6189, pp. 218–232. Springer, Heidelberg (2010)
3. Lew, P., Olsina, L.: Instantiating Web Quality Models in a Purposeful Way. In: Auer, S., Díaz, O., Papadopoulos, G.A. (eds.) ICWE 2011. LNCS, vol. 6757, pp. 214–227. Springer, Heidelberg (2011)
4. Olsina, L., Papa, F., Molina, H.: How to Measure and Evaluate Web Applications in a Consistent Way. In: Modelling and Implementing Web Applications, ch. 13, pp. 385–420. Springer HCIS (2008)