The Journal of Systems and Software 83 (2010) 2015–2018

Contents lists available at ScienceDirect

The Journal of Systems and Software

journal homepage: www.elsevier.com/locate/jss

Editorial

Interplay between usability and software development

Keywords:
Usability
Software development

1. On software development and usability evaluation

Software development is highly challenging. Despite many significant successes, several software development projects either fail completely or produce software with serious limitations. One of the most common limitations is poor usability. Usability is defined as the effectiveness, efficiency and satisfaction with which users achieve their particular goals in a particular context (ISO, 2006). Poor usability may have undesirable effects on software systems, such as a lack of usefulness (i.e., the system does not adequately support the user's core tasks), a lack of productivity gains, or even reduced productivity despite heavy investments in information technology.

Broadly speaking, two approaches have been adopted to address usability limitations. The first approach is to incorporate evaluation activities into software development projects in order to measure and improve the usability of the software. More than 30 years of research into Human–Computer Interaction (HCI) have both generated and validated numerous techniques with which to assist software developers in the evaluation of usability (e.g., Nielsen and Mack, 1994; Constantine and Lockwood, 1999; Bevan, 2003; Blandford et al., 2008).

The second approach is based on the significant advances in techniques and methodologies for user interface design that have been achieved during the last decades. The areas of participatory design and user-centred design have worked on improving the usefulness of information technology by focusing on a deeper understanding of how to elicit, analyse and document user needs and translate them into system development specifications (e.g., Greenbaum and Kyng, 1991; Beyer and Holtzblatt, 1998; Bødker et al., 2004).

The Software Engineering (SE) community has recently acknowledged that usability not only affects the design of user interfaces but also the software development as a whole (Abrahão et al., 2008, 2009) and its, efforts are particularly focused on explaining the implications of usability for requirements gathering (Juristo et al., 2007), software architecture design (Bass et al., 2001; Bass and John, 2003), and the selection of software components (Perry and Wolf, 1992).

Despite these efforts to emphasize the importance of usability, the interplay between HCI and SE, and between the activities that they advocate for the undertaking of software development, has been limited. Research in SE and HCI has been done mainly independently of each other without a substantial exchange of results, and little effort has been made to combine the techniques of the two approaches.

The purpose of this special issue of the *Journal of Systems and Software* is to report on the most recent advances with regard to the interplay between usability evaluation and software development. There are hundreds of techniques related to different software development stages (i.e. requirements elicitation, design, and evaluation). In this issue, we focus on evaluation activities. Some of the papers provide a practical description of some of the basic evaluation principles that may assist practitioners to discover some of the well-known usability evaluation methods and/or strategies. Others provide insights into new trends in usability evaluation.

2. The special issue

This special issue of the Journal of Systems and Software has attracted 27 submissions from America, Asia and Europe. Of these papers, we have been able to accept four for the special issue.

The first paper by Asbjørn Følstad and Kasper Hornbæk entitled "*Work-domain knowledge in usability evaluation: Experiences with Cooperative Usability Testing*" advocates that, when building interactive systems for specific work-domains, a constructive interplay between software development and usability evaluation depends on the availability of work-domain knowledge during the evaluation. The authors then explore the use of the Cooperative Usability Testing method for the evaluation of this kind of systems. In this method, the test participants are actively engaged in the interpretation of their own interaction with the system. Specifically, the study looks at how the test participants' interpretations affected the output of the evaluation and the software developers' subsequent priorities.

The second paper by Bettina Biel, Thomas Grill and Volker Gruhn entitled "Exploring the Benefits of the Combination of a Software Architecture Analysis and a Usability Evaluation of a Mobile Application" proposes combining an inspection method (Software ArchitecTure analysis of Usability Requirements realization – SATURN) with a user test with the aim of identifying what problems are influenced by the software architectures underlying a mobile application early in the lifecycle. This study analyses whether the two usability evaluation methods are complementary by applying them to the same artefacts (software architecture specifications) and comparing the type of usability problems detected by each method.

0164-1212/$ – see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.jss.2010.05.080

The third paper by Anirudha Joshi, NL Sarda and Sanjay Tripathi entitled "*Measuring Effectiveness of HCI Integration in Software Development Processes*" proposes two metrics for measuring effectiveness of HCI integration in software development processes. The first metric (UGAM) is a product metric that measures the extent to which the design of a software product achieves its user experience goals. The second metric (IoI) is a process metric that measures the extent of the integration of HCI activities into SE processes. Both metrics were empirically validated in three independent studies involving students and industry professionals. The results show that the two metrics correlated well with each other and that IoI is a good predictor of UGAM.

Finally, in the fourth paper by Jeff Winter and Kari Rönkkö entitled "*SPI success factors within product usability evaluation*", the authors report eight years of experience in product related usability testing and evaluation with principles for software process improvement. The authors specifically provide practical recommendations and lessons learned regarding usability evaluation in a specific software development context, which may encourage and inspire practitioners to improve their evaluation processes.

In summary, the papers in this issue show some current efforts that combine software development techniques with usability evaluation methods from the field of HCI. The papers cover different aspects of the software development lifecycle, including requirements gathering, architectural design and user interface design. The papers also analyse the quality of certain usability evaluation methods, what types of problem each method detects and whether the methods are complementary. Finally, the papers focus on the role that usability evaluations can play in both product and process improvement.

3. New topics and trends

The papers submitted to this special issue focus on the current advances in usability and software development, reflecting current ideas in the field. However, recent advances in mobile, ubiquitous, social, and tangible computing technologies have moved HCI into practically all areas of human activity. This has led to a shift away from usability engineering to a much richer scope of User Experience (UX), in which user's feelings, motivations, and values are given as much, if not more, attention as ease of use, ease of learning and basic subjective satisfaction (i.e. the three traditional usability metrics).

In order to accommodate this shift, HCI and SE approaches need to respond in a manner that is sensitive to increasingly diverse use contexts, user goals and roles, and new interaction styles (Bødker, 2006). A range of emergent design and evaluation approaches such as experience-centred design (e.g., Blythe et al., 2006), worth-centred design (e.g., Cockton, 2008), and ethnography-informed design (e.g., Dourish, 2006) have been developed. These new approaches deal with issues such as emotion, affect, aesthetics and longitudinal user-artefact relationships that entail the augmentation of certain maturing usability models and methods (Law et al., 2008). Among others, three challenges engendered by the new focus of UX are particularly relevant to software development: the definition of UX; the modelling of UX; the selection and application of UX evaluation methods and measures. Since each of them needs quite a lot of space for their elaboration, here we simply highlight the main arguments involved.

The concept of UX is commonly understood to be *subjective, context-dependent* and *dynamic* (Law et al., 2009), and these key attributes do not appear to be conducive to measurability. In contrast, the formal definition of UX issued by ISO 9241-210: 2010 – *A person's perceptions and responses that result from the use and/or anticipated use of a product, system or service*–suggests that UX can be measured in a manner similar to that of the behavioural and attitudinal usability metrics. Different attempts have been made to demarcate or even dismiss the boundary between usability and user experience at both the conceptual and operationalization levels. A significant implication of this definitional issue is what can be considered as valid measures of UX, which enable professionals to benchmark competitive design artefacts and to select the right design options.

Modelling users' experience, as a basis for producing design guidance, is particularly important. First, measurement models are required to provide a sound basis for UX measures with desirable properties (e.g., reliability, validity, sensitivity). Second, structural models are needed for the purpose of understanding, predicting and reasoning about UX processes with consequences for software design. Despite some visible progress (e.g., Hassenzahl, 2004), a number of issues pertaining to UX modelling are yet to be resolved (Law and Schaik, 2010). Furthermore, it is extremely important to develop practical guidelines for the selection of evaluation methods (e.g., Roto et al., 2009) and an associated set of measures to meet requirements which are specific to the context of interest. It would be helpful to construct a taxonomy of UX qualities with clear definitions grounded in theories. Presumably, the task of constructing such guidelines will only come to fruition when UX research becomes more mature.

In order to facilitate the integration of user experience, we have identified the need for new resources (e.g., theoretical frameworks on the nature of user interactions, metrics) with which to increase the value of usability practice in software development. Although few would argue against making software products usable, actually doing so in practice has proved to be a challenging endeavour.

Acknowledgements

We would like to thank the authors of all the submitted papers for their efforts. In addition, we would like to thank the Editor-in-Chief Hans van Vliet for his support and guidance during the realization of this special issue. Finally, we would like to thank the following colleagues for their help in reviewing the submitted papers:

- Silvia Acuña, Universidad Autónoma de Madrid, Spain
- Samuel Adesoye Ajila, Carleton University, Canada
- Luciano Baresi, Politecnico di Milano, Italy
- Michael Bach Pedersen, KMD, Denmark
- Len Bass, Software Engineering Institute, USA
- Nigel Bevan, Professional Usability Services, UK
- Larry Bernstein, National Software Council, USA
- Marco Brambilla, Politecnico di Milano, Italy
- Cristina Cachero, University of Alicante, Spain
- Tiziana Catarci, Università degli Studi di Roma 'La Sapienza', Italy
- Ricardo Chalmeta, Universitat Jaume I, Spain
- Gan Deng, Automated Trading Desk, USA
- Torgeir Dingsoyr, SINTEF, Norway
- Maria Francesca Costabile, Universita' di Bari, Italy
- Yannis Dimitriadis, University of Valladolid, Spain
- Xavier Ferré, Universidad Politécnica de Madrid, Spain
- Eelke Folmer, University of Nevada, USA
- Asbjørn Følstad, SINTEF, Norway
- Peter Forbrig, Universität Rostock, Germany
- Denis Gillet, École Polytechnique Féderale de Lausanne, Switzerland
- Ilona Heldal, Chalmers University of Technology, Sweden
- Guilherme Horta Travassos, COPPE/UFRJ, Brazil
- Morten Hertzum Roskilde University, Denmark
- Rune Th. Hoegh, ETI, Denmark

Download English Version:

https://daneshyari.com/en/article/458896

Download Persian Version:

https://daneshyari.com/article/458896

Daneshyari.com