Usability measurement and metrics: A consolidated model

Ahmed Seffah · Mohammad Donyaee · Rex B. Kline · Harkirat K. Padda

Abstract Usability is increasingly recognized as an important quality factor for interactive software systems, including traditional GUIs-style applications, Web sites, and the large variety of mobile and PDA interactive services. Unusable user interfaces are probably the single largest reasons why encompassing interactive systems – computers plus people, fail in actual use. The design of this diversity of applications so that they actually achieve their intended purposes in term of ease of use is not an easy task. Although there are many individual methods for evaluating usability; they are not well integrated into a single conceptual framework that facilitate their usage by developers who are not trained in the filed of HCI. This is true in part because there are now several different standards (e.g., ISO 9241, ISO/IEC 9126, IEEE Std.610.12) or conceptual models (e.g., Metrics for Usability Standards in Computing [MUSiC]) for usability, and not all of these standards or models describe the same operational definitions and measures. This paper first reviews existing usability standards and models while highlighted the limitations and complementarities of the various standards. It then explains how these various models can be unified into a single consolidated, hierarchical model of usability measurement. This consolidated model is called Quality in Use Integrated Measurement (QUIM). Included in the QUIM model are 10 factors each of which corresponds to a specific facet of usability that is identified in an existing standard or model. These 10 factors are decomposed into a total of 26 sub-factors or measurable criteria that are further decomposed into 127 specific metrics. The paper explains also how a consolidated model, such as QUIM, can help in developing a usability measurement theory.
1. Introduction

Several studies have reported the benefits of a strong commitment to usability in the software development lifecycle (e.g. Mayhew, 1999; Landauer, 1995). Among the observable benefits of usable user interfaces, one can mention human productivity and performance, safety and commercial viability. Usability is important not only to increase the speed and accuracy of the range of tasks carried out by a range of users of a system, but also to ensure the safety of the user (Repetitive Strain Injury etc.). Productivity is also imperative where the software is used to control dangerous processes. Computer magazine software reviews now include ‘usability’ as a ratings category. The success of commercial software may hinge on these reviews, just as the success of any software relies on the attitude of its users. Attitudes can be influenced by abstract factors such as the look and feel of a product, and how the software can be customized by the user (e.g., colors, fonts, commands).

This explains the increasing numbers of publications in the literature have addressed the problem of how to measure software usability. Several different standards or models for quantifying and assessing usability have been proposed within the Human-Computer Interaction (HCI) and the Software Engineering (SE) communities. Examples of the latter include the ISO/IEC 9126-1 (2001) standard, which identifies usability as one of six different software quality attributes; the ISO 9241-11 (1998) standard, which defines usability in terms of efficiency, effectiveness, user satisfaction, and whether specific goals can be achieved in a specified context of use; and Directive 90/270/EEC of the Council of the European Union (1990) on minimum safety and health requirements for work with computers. However, usability has not been defined in a consistent way across the standards just mentioned or other models described momentarily. Most of these various definitions or models do not include all major aspects of usability. They are also not well integrated into current software engineering practices, and they often lack computer tool support, too.

One consequence of these weaknesses is that perhaps most software developers do not apply correctly any particular model in the evaluation of usability. This is not surprising given that there are few clear guidelines about how various definitions of usability factors, rules, and criteria are related (if at all) and how to select or measure specific aspects of usability for particular computer applications. Instead, actual practice tends to be ad hoc such that developers may employ usability methods with which they are familiar. These choices may not be optimal in many cases. That is, the effort to measure usability may be wasted without a consistent and consolidated framework for doing so. Other motivations to outline a consolidated model for usability measurement are to:

- Reduce the costs of usability testing by providing a basis for understanding and comparing various usability metrics
- Complement more subjective, expert-based evaluation of usability
- Provide a basis for clearer communication about usability measurement between software developers and usability experts
- Promote sound usability measurement practices that are more accessible to software developers who may not have strong backgrounds in usability engineering