Proceedings of the Federated Conference on Computer Science and Information Systems pp. 1005–1008 DOI: 10.15439/2018F137 ISSN 2300-5963 ACSIS, Vol. 15

Usability attributes revisited: a time-framed knowledge map

Paweł Weichbroth
Faculty of Management, WSB University in Gdansk,
ul. Grunwaldzka 238A, 80-266 Gdansk, Poland
Email: pawel.weichbroth@hotmail.com

***Abstract*—Software usability plays a major role in the quality perceived by its users. However, a variety of definitions and associated attributes shows that there is still no consensus in this area. The overall purpose of this paper is to present the results of a critical and rigorous literature review, the aim of which is to demonstrate all the relevant usability definitions and related attributes introduced till now. This comprehensive view, depicted by a time-framed knowledge map, provides an in-depth understanding of the observed evolution on the one hand, and also serves as a guide for usability engineers to address some non-functional requirements, on the other.**

I. Introduction

DISCUSSING usability in general is one thing, but explaining its nature in detail in an applicative manner is another. An ordinary user of any application or device would point to ease of use when asked about the primary attribute of its usefulness. Indeed, in an article for Time Magazine, Tim Bajarin, while discussing "*6 Reasons Apple Is So Successful*", placed this answer in second place [O1]. Beyond this, market success is the hard work of not only the sales department, but mostly, visionary usability engineers.

To design and produce usable software, the concept of usability must be understood. However, till now, a plethora of definitions and related attributes, on the one hand, do not make it easy, while on the other, a flood of buzz words make it even worse. To fill this gap, we investigate existing models and standards to find answers to the research questions: (RQ_1) what is software usability? and (RQ_2) which attributes most frequently contribute to the software usability?

The reasons for performing this study are twofold. Usability is a major concern in every stage of the design process, and now design thinking is regarded as a system that fosters innovation. Secondly, in face of the observable shift from desktop to mobile application, nowadays, usability again captivates the interest of software vendors in adapting their products to the new settings.

The rest of the paper is organized as follows. In Section II, the research method is depicted, followed by the recognition and definition of usability attributes identified in multiple standards and models (Section III). Finally, we conclude the paper and point out future research directions (Section IV). The paper's contributions are: outlining past research, highlighting its drawbacks and providing one synopsis in the form of a time-framed knowledge map (Table I).

II. Research method

In order to provide the relevant answers to the research questions, the literature survey was based on the approach of Webster and Watson [O2]. A rigorous search process was applied to the existing body of knowledge to identify norms, scientific papers and books, as well as technical reports that contribute to the subject field. The term "rigorous" stands for the reliability and validity of the search process. The former refers to the extent to which its outcome is consistent over time and an accurate representation of the total population under study, and if the results of the study can be reproduced under a similar procedure. The latter is the degree to which the literature search accurately uncovers the relevant sources. The research was composed of three stages, namely, literature search, literature analysis and selection of attributes.

The literature search embraced databases of widely recognized publishers whose scopes correspond to computer science, information systems or similar (e.g. ergonomics). In particular, ACM, Elsevier, IEEE and Springer were searched for the keywords *usability* and *software usability*, followed by a search of aggregated databases that store records of numerous publishers, namely EBSCOhost, Scopus and Web of Science. An electronic search was performed against the metadata of all the publications. Next, only articles published in relevant journals or books were taken into account. However, the abundance of search results pushed us to redefine our query by limiting the search fields to the title, keywords and abstract. Although this search was not exhaustive, it submitted a comprehensive input.

The literature analysis aimed to identify adequate content for the research questions (see Section I). To achieve this goal, firstly, the publications were reviewed to eliminate those biased by a context (user-specific attributes such as: age, occupation, sex or system-specific support features like visually impaired, disability), unrelated to computer systems or tailored to specific project constraints. Secondly, the remaining publications were read completely with a focus on their parts devoted to the subject, in order to catalog valid data.

The selection of attributes stage was based on the following, mutually nonexclusive criteria, applied to all the classified items: (*a*) published in English, (*b*) related to usability studies, (*c*) published by an international standardsetting body or governmental institution, (*d*) referenced in research papers. The 'charting' technique was adopted to synthesize and interpret gathered qualitative data by sifting, charting and sorting information chunks according to key issues and themes. The final outcome is twofold: a narrative review of the recent research, and a time-framed knowledge map (Table 1) as the research summary.

III. Usability Attributes

Among efforts to explain what the term means, Shackel claims that the definition of usability was probably first attempted by Miller in 1971 in terms of measures for "ease of use" [1].

In 1977, McCall defines usability as "the effort required to learn, operate, prepare the input and interpret the output of a program" [O3]. However, the fact that it was already included as a quality factor did not really imply attention from software engineers, but it was regarded as an extension of general data processing system design. This perspective was typical of its time.

From the late 1970s, through the 1980s, it was in fashion to discuss software and other hardware-based artefacts which were easy to use as being "user-friendly". The question of "what is user-friendly" had little credibility; however, the problem of the efficiency of computer programs was often deliberated. These two issues were recognized together by several researchers, including Bennet (1984) and Shackel (1986), who introduced comparable definitions [O4, O5].

Shackel framed usability in terms of the system: *effectiveness*, the *easiness to learn*, its *flexibility* and user *attitude* [O5]. A formal definition, established in 1991, says that it is "the capability in human functional terms to be used easily and effectively by the specified range of users, given specified training and user support, to fulfil the specified range of tasks, within the specified range of environmental scenarios" [1]. The notion is next split into four attributes: *effectiveness*: "in terms of performance (e.g. time, errors, number of sequence activities) in learning, relearning and carrying out a representative range of operations", *learnability*: "within a specified time from the installation and start of user training" and "the amount of training and user support", *flexibility*: which allows "adaptation to some specified percentage variation in tasks and/or environments beyond those first specified" and *attitude*: "within acceptable levels of human cost in terms of tiredness, discomfort, frustration and personal effort". However, despite the interest that has been aroused around this approach, it appears to have limitations. Flexibility is particularly difficult to specify, communicate and test in a real system development environment. Among others, Preece et al. [O6] also draw heavily on its rationale. Today, flexibility is seldom considered explicitly.

In 1987, the FURPS model was first introduced by Grady and Caswell [2]. In 1992, the original model "was extended to empathize various specific attributes" and re-designated to FURPS+ in which usability is described by four attributes: *aesthetics, consistency*, *documentation* and *human factors* [3].

In response to the need of the software industry to standardize the evaluation of software products using quality models, in 1991, the ISO Organization issued a standard, namely ISO 9126, that specifies six areas of importance for software evaluation, including usability, defined as "a set of attributes of software which bear on the effort needed for use, and on an individual assessment of such use, by a stated or implied set of users" [4]. This standard puts forward three attributes (named as sub-characteristics): *understandability*, *learnability*, *operability* (see also: [O7] and [O8]).

In 1993, Kirakowski and Corbett presented the SUMI (Software Usability Measurement Inventory) method to measure users' perception of software usability, providing three different layers of output. The second layer has five sub-scales: *affect*, *efficiency*, *learnability*, *helpfulness* and *control* [5].

Nielsen (1993) associates usability with five attributes: *learnability*, *efficiency*, *memorability*, *errors* and *satisfaction* [6]. Every definition starts with words "the system should be or have" which, in particular, applies to its capability of being easy to learn, efficient to use, easy to remember, free of errors and pleasant to use. Nielsen emphasizes learnability as "the most fundamental attribute of usability, since most products need to be easy to learn, and since the first experience that most people have with a new product is that of learning to use it", and also relates it to novices' ability to reach a reasonable level of performance rapidly, which indicates the direct relation between learnability and efficiency, i.e. the user interface should be easy to learn so that the user is able to complete a given task successfully in a certain time.

ISO 9241-11 (1993, 1998) [7, 10], along with Bevan (1995) [9], consider *effectiveness*, *efficiency* and *satisfaction* as usability measures. These standards relate to usability as a high level quality objective, which is reflected by its definition: "the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use". Today, this is the most recognizable usability definition.

IBM's CUPRIMDSO (1994) quality assurance system includes customer *satisfaction* only, with a five-point scale for evaluating the value of the product that results from its effective usage [8].

Constantine and Lockwood (1999) propose five facets, as different aspects of a system and its user interface that contribute to usability: *learnability*, *rememberability*, *efficiency in use*, *reliability in use* and *user satisfaction* [11]. The authors declare that software which "leads its users to make fewer mistakes will be more reliable in use", that is, "in how it functions in combination with its users and in how it promotes reliable human performance". Reliability in use is more closely tied with the user interface design than with coding and debugging.

Ten years after the introduction of ISO/IEC 9126:1991, the standard was refined by a group of software engineer experts to ISO/IEC 9126:2001. Even though it does not cover all aspects of software quality from the product perspective, it is still the most comprehensive model developed to date. Here, usability is specified as "the capability of the software product to be understood, learned and liked by the user, when used under specified conditions", construed as five attributes: *understandability*, *learnability*, *operability*, *attractiveness* and *usability compliance* [12].

Abran et al. (2003) developed an Enhanced Usability Model, by integrating process-related (ISO 9241) and product-related (ISO 9126) standards, which includes five attributes: *effectiveness*, *efficiency*, *satisfaction*, *learnability* and *security* [13]. To justify the last one, they provide a list of five arguments, particularly referring to ITSEC (Information Technology Security Evaluation Criteria) and to three normative standards (IEC 300, ISO 13407: 1999, ISO/IEC 9126), and eventually closing by indicating that "security is a characteristic of CHI, which is particularly important in an industrial context".

Seffah et al. (2006) elaborate the QUIM model by consolidating ten attributes (originally named factors): *efficiency*, *effectiveness*, *productivity*, *satisfaction*, *learnability*, *safety*, *trustfulness*, *accessibility*, *universality* and *usefulness*, each of which corresponds to a specific facet of usability, identified in an existing standard or model [14]. These ten factors are decomposed into a total of 26 sub-factors or measurable criteria that are further broken down into 127 specific metrics. It seems that the novelty of this work brings into view a complete and legitimate instrument to evaluate usability.

In ISO 9241-210:2010 [15], the usability definition is adapted from ISO 9241-11:1998 [10], along with its set of attributes as well.

The ISO 25010:2011 standard on quality models updates

TABLE I.
USABILITY ATTRIBUTES OF VARIOUS STANDARDS AND MODELS

Attribute / [Ref. No] & Year	**[1] 1991**	**[2, 3] 1987**	**[4] 1991**	**[5] 1993**	**[6] 1993**	**[7] 1993**	**[8] 1994**	**[9] 1995**	**[10] 1998**	**[11] 1999**	**[12] 2001**	**[13] 2003**	**[14] 2006**	**[15] 2011**	**[16] 2016**	**[17] 2018**
Effectiveness	●					●		●	●			●	●	●		●
Learnability	●		●	●	●					●	●	●	●		●	
Flexibility	●															
Attitude	●[2]															
Aesthetics		●														
Consistency		●														
Documentation		●														
Human factors		●														
Understandability			●								●				●[1]	
Operability			●								●				●	
Affect				●												
Efficiency				●	●	●		●	●	●[1]		●	●	●		●
Helpfulness				●												
Control				●												
Memorability					●					●[1]						
Errors					●										●[1]	
Satisfaction					●	●	●	●	●	●[1]		●	●	●		●
Reliability in Use										●						
Attractiveness											●				●[1]	
Usability Compliance											●					
Security												●				
Productivity													●			
Safety													●			
Trustfulness													●			
Accessibility													●		●	
Universality													●			
Usefulness													●			

●: Included. [1]: Defined under a similar name. [2]: Attitude has a moderate connotation contrary to satisfaction, thus they are not combined.

and brings together previous standards, defining three views of quality: internal quality, external quality and quality in use [16]. Of particular interest in ISO 25010 is the standard's new breakdown of quality in use and usability. The former is defined as "the degree to which a product or system can be used by specific users to meet their needs to achieve specific goals with effectiveness, efficiency, freedom from risk, and satisfaction in specific contexts of use", while the latter is defined the same as in [10, 15]. Usability can either be specified or measured as a product quality characteristic in terms of its subcharacteristics, or specified directly by measures that are a subset of quality in use [O8]. The standard delineates its six attributes (subcharacteristics) [16]:

– *appropriateness recognizability*: degree to which users can recognize whether a product or system is appropriate for their needs;
– *learnability*: degree to which a product or system can be used by specified users to achieve specified goals of learning to use the product or system with effectiveness, efficiency, freedom from risk, and satisfaction in a specified context of use;
– *operability*: degree to which a product or system has attributes that make it easy to operate and control;
– *user error protection*: degree to which a system protects users against making errors;
– *user interface aesthetics*: degree to which a user interface enables pleasing and satisfying interaction for the user;
– *accessibility*: degree to which a product or system can be used by people with the widest range of characteristics and capabilities to achieve a specified goal in a specified context of use.

Compared with the 2001 edition (ISO/IEC 9126-1) [12], two new subcharacteristics were introduced: *user error protection* and *accessibility*, while usability compliance was withdrawn. It is worth noting that understandability was renamed to appropriateness recognizability, along with attractiveness to user interface aesthetics [16].

Similarly (see: ISO 9241-210:2010 and ISO 25010:2011), in ISO 9241-11:2018 [17] the usability definition is adapted from [15] together with its three attributes.

IV. Conclusions

Table 1 shows the final list of all usability attributes, together with the representative references, including also standards from the International Organization for Standardization (ISO), covering usability in human-computer interaction. From the list, the most frequent are *efficiency* and *satisfaction* (each supported by 10 references), *learnability* (9) and *effectiveness* (8). The least frequent are *understandability* and *operability* (3), *memorability*, *errors*, *attractiveness* and *accessibility* (2), while the rest attributes occur only once. The next research step is to determine usability facets relevant for mobile applications.

References

[1] B. Shackel, Usability – Context, framework, definition, design and evaluation. in B. Shackel and S. Richardson (Eds.), *Human Factors for Informatics Usability*, Cambridge 1991, pp. 21–38.

[2] R. B. Grady and D. Caswell, *Software metrics: Establishing a company-wide program*. Englewood Cliffs: Prentice-Hall 1987.

[3] R. B. Grady, *Practical software metrics for project management and process improvement*. Englewood Cliffs, NJ: Prentice-Hall 1992.

[4] International Organisation for Standardisation, ISO/IEC 9126. *Information Technology – Software Product Evaluation – Quality Characteristics and Guidelines for Their Use*. Genève 1991.

[5] J. Kirakowski and M. Corbett, SUMI: *the software usability measurement inventory*. British Journal of Educational Technology, Vol. 24(3), 1993, pp. 210-212.

[6] J. Nielsen, *Usability Engineering*, Academic Press, London 1993.

[7] International Standards Organization, ISO DIS 9241-11. *Ergonomic requirements for office work with visual display terminals. Part 11: Guidance on usability*. London 1993.

[8] D. L. Bencher, *Programming quality improvement in IBM*. Technical forum. IBM Systems Journal, 33(1), 1994, pp. 215–219.

[9] N. Bevan, *Measuring usability as quality of use*. Software Quality Journal, 4, 1995, pp. 115–130.

[10] International Organization for Standardization, ISO: 9241-11:1998. *Ergonomic requirements for office work with visual display terminals (VDTs), Part 11: Guidance on usability*. Geneva 1998.

[11] L. L. Constantine and L. A. D. Lockwood, *Software for Use: A Practical Guide to the Models and Methods of Usage-Centered Design*. Addison-Wesley, New York 1999.

[12] International Organization for Standardization/International Electrotechnical Commission, ISO/IEC 9126 *Standard, Software Engineering, Product Quality*. Part I & Part IV. Geneva 2001.

[13] A. Abran, A. Khelifi, W. Suryn, and A. Seffah, *Usability meanings and interpretations in ISO standards*. Software Quality Journal, 11(4), 2003, pp. 325–338.

[14] A. Seffah, M. Donyaee, R. B. Kline and H. K. Padda, *Usability measurement and metrics: A consolidated model*. Software Quality Journal, 14(2), 2006, pp. 159–178.

[15] International Organization for Standardization, ISO 9241-210:2010. *Ergonomics of human-system interaction -- Part 210: Human-centred design for interactive systems*. Geneva 2010.

[16] International Organization for Standardization, ISO 25010:2011. *Systems and software engineering -- Systems and software Quality Requirements and Evaluation (SQuaRE) -- System and software quality models*. Geneva 2011.

[17] International Organization for Standardization, ISO 9241-11:2018. *Ergonomics of human-system interaction -- Part 11: Usability: Definitions and concepts*. Geneva 2018.

Other References

[O1] T. Bajarin, *6 Reasons Apple Is So Successful*. Time. May 07, 2012.

[O2] J. Webster, and R. T. Watson, R. T., Analyzing the past to prepare for the future: Writing a literature review. MIS quarterly, 2002, xiii-xxiii.

[O3] J. A. McCall, P. K. Richards, G. F. Walters, *Factors in Software Quality*. US Rome Air Development Center Reports, Vol I, II, III. US Department of Commerce, USA, 1977.

[O4] J. L. Bennett, Managing to Meet Usability Requirements: Establishing and Meeting Software Development Goals. in J. L. Bennett, B. Case, J. Sandelin and M. Smith (Eds.), *Visual Display Terminals: Usability Issues & Health Concerns*. Prentice-Hall 1984, pp. 161–184.

[O5] B. Shackel, Ergonomics in Design for Usability. in M. D. Harrison and A. Monk (Eds), *People and Computers: Designing for Usability*. Proc. of Sec. HCI Conference of the BCS, York 1986, pp. 45–64.

[O6] J. Preece, Y. Rogers, H. Sharp, D. Benyon, S. Holland and T. Carey, *Human-Computer Interaction*. Pearson, 1994.

[O7] A. Przybyłek, *An empirical study on the impact of AspectJ on software evolvability*. Emp. Soft. Eng., 23(4), pp. 2018–2050, 2018.

[O8] P. Weichbroth, *Delivering Usability in IT Products: Empirical Lessons from the Field*. International Journal of Software Engineering and Knowledge Engineering, 28(07), 1027-1045, 2018.