Research questions on integrating user experience approaches into software development processes

Bálint Szabó, Károly Hercegfi
Department of Ergonomics and Psychology
Budapest University of Technology and Economics
Budapest, Hungary
{szabobalint, hercegfi}@erg.bme.hu

***Abstract*— In the narrowly interpreted software development processes, most companies use already proven, partially standardized rivalling models. However, these programming-centered development models do not fit very well with the more broadly interpreted models of product management processes. This article provides a systematic overview of current issues in research focusing on the human aspects: in what way cognitive processes can co-evolve with software and infocommunications devices.**

Keywords* — *user experience, human factors, software development, UX maturity

I. Introduction

From the beginning, the field of the human-computer interaction (HCI) area was defined as a multidisciplinary scientific discipline. This resulted in an explicit recognition in the 80's, that distinct perspectives and competencies are required for making a functional, usable, universally accessible computer system to humans. For this purpose, qualified experts in computer science and software engineering areas started to work together with other specialists, who had knowledge in behavioral sciences to create interdisciplinary teams as essential criteria for usable systems. One promising new approach is the cognitive infocommunications, which has emerged as a synergic combination of the research areas of infocommunications and cognitive sciences [1, 2]. Another, more wide-spread interdisciplinary phenomenon is user experience (UX), however its definition is not unified yet [3]. According to the rising of these approaches, nowadays, the appearance of user-centered design techniques is quite important to develop usable graphical user interface in interactive infocommunication systems with adequate user experience. However, the biggest problem is that the importance of this recognition is not yet reflected in software engineering methods. [4].

II. Background for software development

A. Background for software development

The process of software development has taken different routes at different times in the past, which effects the adaptability of the user-centered aspects and applied methodologies as well. In practice, there are a plenty of traditional, idealized software development models for the companies, which can be grouped into four basic categories according to Mohapatra. There are the code and fix, the waterfall, the evolutionary and the spiral variations, that need to be distinguished. [5]

In the past, requirements were fully known, so the development processes primarily involved only coding and fixing activities. At that time, the projects were typically connected to a science or an engineering application and the developer was also the user of the software [6]. As time passed, this code and fix typed model turned out to be highly inadequate, so the penetration of the information technology and the changes in the software development environment needed a more systematic way to develop software products.

Influenced by the development processes in the famous air defense software project (called SAGE) Benington [7], Rosove [8] and Royce [9] proposed the waterfall model of the software development process. In this model, the development phases are connected to each other in a cascade sequence, where the following step can only be started if the previous individual phase is completed [10]. This development solution has become the basis for most software acquisition standards for a few years in government and industry, but its biggest disadvantage was the emphasis on fully elaborated documents as completion criteria for almost every phase (mostly in requirement analysis and system design, which does not work well during the development of interactive end-user software). [11]

Therefore, the evolutionary model allows concurrent activities and quick feedback by developing a working model of the software, that could be presented to the customer to create system with the appropriate functionality [12]. This method can be implemented using the incremental approach or prototyping. The spiral model illustrates the software development process as a spiral (a series of non-consecutive activities). The spiral represents the core development activities of the software process in each round. This was the first approach with more emphasis placed on risk analysis. [13]

So in the past decade, a number of new ideas have emerged on the software development processes. All of these new, non-traditional development approaches (like component-based software development [14], win-win spiral model [15], or rational unified process [16]) are iterative and incremental with an extra view to complying with changing user requirements, but lately, the most widespread method is the agile development process. This trend is practically a response to the

978-1-5386-1264-4/17/$31.00 ©2017 IEEE

accelerating technology and market development. The new methodology and its tools (such as Scrum, Kanban, or eXtreme Programming) are much more focused on market changes, continuous communication with customer and feedback on how to handle the development direction more flexible than earlier models. [17]

B. Software quality standards

In recent years, software have become extremely important for the companies from a business point of view, so software quality was of great interest for stakeholders. Nowadays, software products should have a great level of user convenience, quality and utility too. In order to meet these new requirements, there are different ISO standards and certifications that are accepted by customers [18]. In software engineering, there is always an intention for finding better techniques and methods to ensure the best quality at reasonable cost, driven by external market forces. For example, the system and software quality model of ISO/IEC 25010:2011 has eight attributes defined for measuring the quality in a structured set of characteristics and sub-characteristics: functional stability, performance efficiency, compatibility, usability, reliabilityy, security, maintainability, portability. (The predecessor of this standard was the ISO/IEC 9126:1991, and it defined six similar dimensions.) These attributes can be used in the assessment of the quality of the software, but they themselves do not give guidance for the developing processes.

There are a plenty of other ISO standards that could be used by companies. The quality management system requirements are defined in the ISO 9001 standard. The main goal of this document is to satisfy customer needs, which is a basis for developing a quality software product [19]. Another regulation for software lifecycle processes is the ISO/IEC 12207 international standard, which defines all the tasks required for developing and maintaining software [20]. Furthermore, there is the Software Process Improvement and Capability Determination standard (ISO/IEC 15504), which gives technical documents for computer software development processes and related business management functions. [21]

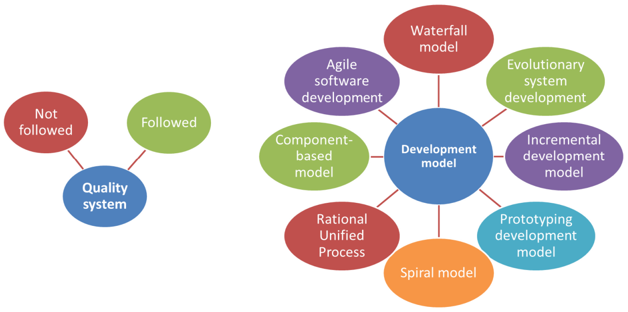

Fig. 1. Development methods and quality systems in the life of the software companies

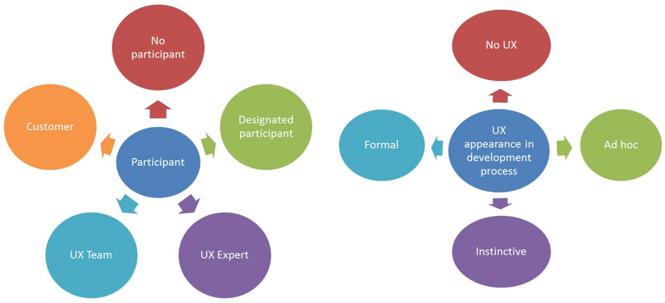

Fig. 2. The appearance of the UX aspects in the life of software companies

III. Research Questions

Although software companies can be grouped by many predefined criteria (like the company formation, size, income, type and function of the produced software), in this research it is more relevant, that what major groups can be distinguished according to the development models and the quality systems applied by them (Fig. 1).

Another central research question is how the user-centered aspects appear during the development processes in the life of the software companies; which participant of the organization is involved (Fig. 2).

In recent years, user experience design and human factors have become key components in many business models, but there are still technology companies for which these factors do not play a central role in the organization's life. The lack of taking into account the human factors is a serious problem in connection with the wide acceptance and application of different user-centered methodologies. For this reason, there are models developed to determine the maturity of organizations along different subdimensions associated with UX. In this model, the six key indicators of an organization’s UX maturity are the followings [22]:

(1) UX involvement timing in the design and development process.

(2) UX competence and connecting resources in the organization or the ability to bring those in as quickly as needed.

(3) The use of appropriate techniques to obtain and understand user input and adequate evaluation to capture UX design.

(4) UX acceptance (value and necessity) in the leadership and in the whole culture of the company from a business perspective.

(5) The integration degree of UX with other processes to create the user experience of the products by the individuals in the organization.

(6) Design thinking is a natural part of the organization to drive consistent customer experience.

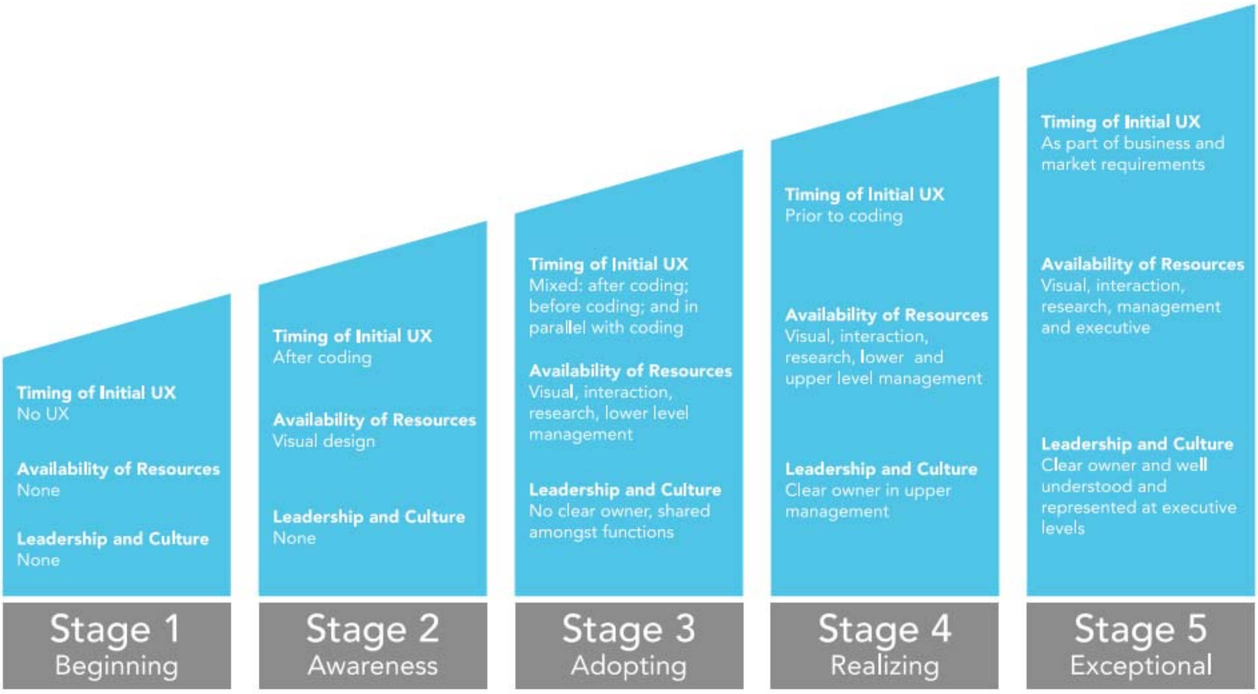
Timing of Initial UX
No UX
Availability of Resources
None
Leadership and Culture
None
Stage 1
Beginning
Timing of Initial UX
After coding
Availability of Resources
Visual design
Leadership and Culture
None
Stage 2
Awareness
Timing of Initial UX
Mixed: after coding; before coding; and in parallel with coding
Availability of Resources
Visual, interaction, research, lower level management
Leadership and Culture
No clear owner, shared amongst functions
Stage 3
Adopting
Timing of Initial UX
Prior to coding
Availability of Resources
Visual, interaction, research, lower and upper level management
Leadership and Culture
Clear owner in upper management
Stage 4
Realizing
Timing of Initial UX
As part of business and market requirements
Availability of Resources
Visual, interaction, research, management and executive
Leadership and Culture
Clear owner and well understood and represented at executive levels
Stage 5
Exceptional

Fig. 3. The stages of UX Maturity Model (showing three of the six key indicators of the UX maturity [22])

Knowing and defining an organization's UX goals and its level of maturity can guide decisions ranging from where an organization seeks help (by hiring UX experts), to what processes they use and implement, and it could be useful in decision making [23]. Thus, the subject of this study is also to determine the level of the different companies according the assessment of UX maturity model.

IV. RESEARCH IN PROGRESS

The aim of the research in progress is to deepen the knowledge of the production process of different types of software products, including the detailed examination of how user-centered aspects are used in software development practice.

A. Methods

The current practice of software companies (applied models, the integrity of methods of usability and user experience) are studied by conducting interviews. A research done on a sample of 15-20 professional leaders is a good starting point for getting to know the real practice of companies and the attitude of the respondents toward the topic.

This comparative, exploratory, qualitative research is carried out with UX experts or (in case of their absence) with development or project leaders.

B. Preliminary results

As a global trend, large companies in the software market are already using agile in development, while the small and medium-sized businesses have recently implemented these methodologies (or these project management tools are just being implemented right now). Agile (developed in the 2000s) is a novel, quality-focused development approach that builds on co-operation.

Although this trend may not necessarily apply in all areas, and cannot be observed for all companies, it well fits especially software companies with the right size, skills and organization maturity. Some comparative examples of these differences are listed below:

There is a company which works with its own methodology in the development of software products and that is uniformly used for all release. In the beginning it meant the use of the waterfall model, which expanded with different iteration phases to move to the agile direction. According to the interviewee, the design is less emphasized in agile methodology, which gives place for more efficient work, but the management is uncertain about the exact outcome. Therefore, the practical solution is to balance the two methodologies during the development to find the optimum between efficiency and predictability.

On the other hand, at another company, developments have been carried out applying only agile. The interviewee cannot imagine working with other methodologies, as he believes it is the only possible way to implement user-centered design, due to traditional models are much less supportive in point of development.

In the sample, there is a company where the users are involved in design, development and testing phase during the product development process, within the framework of an online focus group. The 10-member product manager group dealing with user experience at the organization ensures the continuous interaction with users throughout the development. In another company there is a 3-member professional UX team for this purpose who supports developer's work in predefined areas in order to ensure the user-centered aspects.

Full presentation of the obtained results is possible across multiple dimensions, as there are further differences, for example in regard to the selection of the used methodologies or the number of users involved during development, but of course the size of the group of employees deal with user experience and usability and the tasks they perform are quite different in the various organizations.

C. *Further reasearch*

In addition to the interviews, we also focus on how particular UX methods can be useful in software development processes. As a model for this further research question, we examine the ergonomic quality and usability of the specific software by eye-tracking based empirical series of experiments, proven the cost-effective integration of the method into the development processes by exploring its limitations and possibilities.

In quality standards, the usability of software products and the general dimensions of software testing cannot be measured directly, it can be determined with usability factors (like efficiency, number of errors).

The usability factors and the quantitative data generated during the eye-tracking experiments allow us to measure the usability and user experience quantitatively, thus making it possible to compare the efficiency of standard and eye tracking supported software development processes.

CONCLUSION

According to the preliminary expectations, the development takes place along different steps in every company, which only partially resemble to the software life-cycle models defined in literatures, as they are often supplemented with specific activities and agile elements.

Overview and synthesis of the various UX practices are useful for future decision-makers of different (not only software related) companies, and a systematic overview of the methodologies used in the software market is also novel from a scientific point of view.

In addition, the laboratory research contributes to the applicability of usability studies from economics-related aspects with return calculations.

REFERENCES

[1] P. Baranyi and Á. Csapó, “Definition and synergies of cognitive infocommunications”, Acta Polytechnica Hungarica, vol. 9 (1), pp. 67-83.

[2] P. Baranyi, A. Csapo, G. Sallai, Cognitive Infocommunications (CogInfoCom), Cham: Springer, 2015.

[3] E. L. Law, V. Roto, M. Hassenzahl, A. P. O. S. Vermeeren, and J. Kort, “Understanding, scoping and defining user experience: a survey approach”, Proc. CHI’09, pp. 719-728.

[4] A. Seffah, J. Gulliksen, and M. C. Desmarais, Human-Centered Software Engineering - Integrating Usability in the Software Development Lifecycle, Dodrecht: Springer, 2005.

[5] M. K. J. Pratap, Software Engineering – a Lifecycle Approach, New Delhi: New Age International Publishers, 2009.

[6] C. Ghezzi, M. Jazayeri, and D. Mandrioli, Fundamentals of Software Engineering, New Delhi: Prentice-Hall of India, 1994.

[7] H. D. Benington, “Production of Large Computer Programs”, IEEE Annals of the History of Computing, IEEE Educational Activities Department, 1983, 5 (4), pp. 350–361.

[8] P. E. Rosove, Developing Computer-Based Information Systems, New York, NJ: Wiley, 1967.

[9] W. W. Royce, “Managing the Development of Large Software Systems”, Technical Papers of Western Electronic Show and Convention (WesCon) August 25–28, 1970, Los Angeles, USA.

[10] N. D. Birrell and M. A. Ould: A Practical Handbook for Software Development, London: Cambridge University Press, 1988.

[11] B. W. Boehm, “A spiral model of software development and enhancement”, Computer, vol. 21 (5), pp. 61–72, 1988.

[12] P. J. Denning, C. Gunderson, and R. Hayes-Roth, “Evolutionary system development”, Comm. ACM, 2008, 51 (12), pp 29–31.

[13] N. M. A. Munassar, and A. Govardhan, “A comparison between five models of software engineering”, IJCSI International Journal of Computer Science Issues, 2010, vol. 7 (5), pp. 94-101.

[14] I. Gorton, G. T. Heineman, I. Crnković, H. W. Schmidt, J. A. Stafford, C. Szyperski, and K. Wallnau, Proc. Component-Based Software Engineering, 9th International Symposium, Västerås, Sweden, June 29 - July 1, 2006.

[15] B. Boehm, A Egyed, J. Kwan, D. Port, A. Shah, and R. Madachy, “Using the WinWin Spiral Model: A Case Study”, Computer, 1998, vol. 31 (7), pp. 33-44.

[16] P. Kruchten. The Rational Unified Process An Introduction, 2nd ed., Boston, MA: Addison-Wesley, 2000.

[17] K. Beck, M. Beedle, A. Bennekum, A. Cockburn, W. Cunningham, M. Fowler, J. Grenning, J. Highsmith, A. Hunt, R. Jeffries, J. Kern, B. Marick, R. C. Martin, S. Mellor, K. Schwaber, J. Sutherland, and D. Thomas, “The agile manifesto”, Software Development Magazine, 2001/08, pp. 1-7.

[18] B. Dugalic, and A. Mishev, “ISO Software Quality Standards and Certification”, Proc. BCI’12, Sep 16-20, 2012, Novi Sad, Serbia, pp. 113-116.

[19] Quality management systems: Requirements, ISO 9001:2015, 5th ed.

[20] F. J. Pino, M. T. Baldassarre, M. Piattini, G. Visaggio, and D. Caivano, “Mapping software acquisition practices from ISO 12207 and CMMI”, Proc. ENASE 2008, pp. 234-247.

[21] H. Van Loon, Process Assessment and ISO/IEC 15504: a reference book, Springer Science & Business Media, 2004.

[22] J. Fraser and S. Plewes, “Applications of a UX maturity model to influencing HF best practices in technology centric companies – Lessons from Edison”, Procedia Manufacturing, 2015, vol. 3, pp. 626-631.

[23] L. Chapman and S. Plewes, “A UX maturity model: Effective introduction of UX into organizations”, Proc. International Conference of Design, User Experience, and Usability, 2014, pp. 12-22.