How to apply human-centered design process (HCDP) to software development process?

Shin'ichi fukuzumi1), Yukiko Tanikawa2)
NEC Corporation
1) Data Science research laboratories
2) Business Innovation strategy division
Kawasaki, Japan
1) s-fukuzumi@aj.jp.nec.com,
2) y-tanikawa@cw.jp.nec.com

Natsuko Noda
Shibaura Institute of Technology
College of Engineering and Design
Tokyo, Japan
nnoda@shibaura-it.ac.jp

***Abstract*—This is a position paper of research related to usability and software engineering. Important problem to develop software with high usability is that it is difficult to apply human centered design process to software development process. To solve this, we propose three issues, they are 1) clarification of "demand" and evaluation process about user requirements, 2) how to fill gaps between software engineers and usability experts about HCD, 3) solution of twin peaks problem (gap between requirements and architecture). For these issues, it is necessary to discuss detail targets.** *(Abstract)*

Keywords-Human centered design, usability, development process, software engineer (key words)

I. INTRODUCTION

In 2010, ISO9241-210 "Human-centred design for interactive system" which is an ergonomic related standard about human centered design was published [1]. Discussion related human centered design (HCD) and user experience (UX) which is newly defined in this standard becomes active in IT business field. This standard is put to practical use as an example that shapes HCD concept. Fig. 1 shows the relationship among each HCD activity. As shown in this figure, HCD has six activities. They are,

1) "Plan the human-centered design process" which is a decision phase of a project to apply HCD to a target system,
2) "Understand and Specify the context of use" which is a phase that a project get information how user uses a target system,
3) "Specify the user needs and the user requirements" which is a phase to extract users' needs to a target system and to specify the needs,
4) "Produce design Solution to meet user requirements" which is a phase to make a prototype or production according to the specification,
5) "Evaluate the design against requirements" which evaluates whether the design meets the requirements,
6) "Design solution meet user requirements" which evaluates whether the solution meets requirements".

In these, 2), 3), 4) and 5) are main activities in HCD process. These activities and process are similar to development process. However, as shown in Fig. 2, these activities are applied to each phase in development process. Especially, upper phase in development process is more important for developing product, system or service with high usability.

HCD is a method to give better UX to stakeholders and to provide system and product with high usability for users and stakeholders [2]. UX white paper [3] and Fig. 3 shows that experience includes "before use", "during use", "after use" and "through total usage". "Users want to do" can be verified through experience in each step. When developing usable software and system, it is easy to verify their usability by usability test [4]. However, it is difficult to check whether these products or system achieve that a user wants to really do in which required in UX white paper.

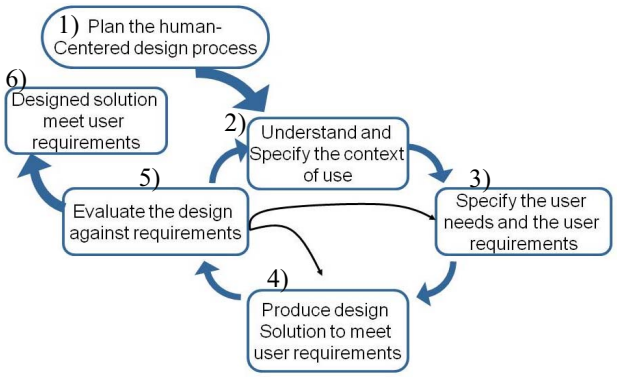

Figure 1. Relationship among each HCD activity [1]

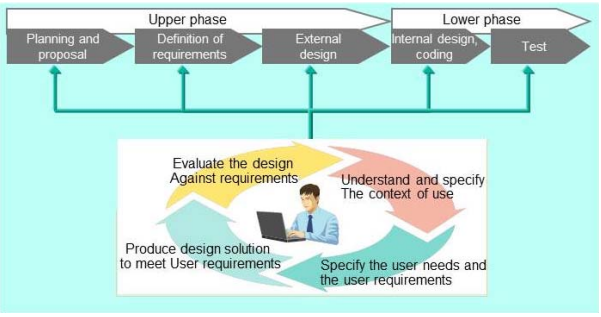

Figure 2. The relationship among HCD activities and development process

978-1-5386-0400-7/17 $31.00 © 2017 IEEE
DOI 10.1109/DISE.2017.10

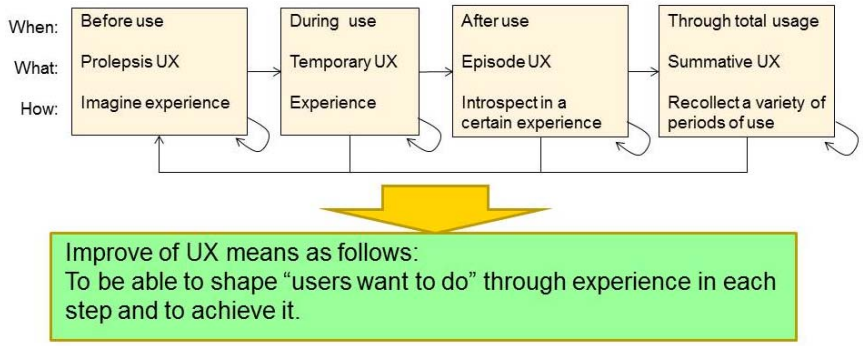

Figure 3. User Experience step [3]

There are many guidelines for realizing high usability. But past studies [5, 6] and field trials of our method [7] revealed that there are still a lot of difficulties for software engineers in design activities considering usability. So, we think it is necessary to develop methods for applying HCD process to software development process.

II. OUR TRIALS

In order to solve this problem, we have tackled from the viewpoints of three approaches shown in below.

A. Ergonomic approach

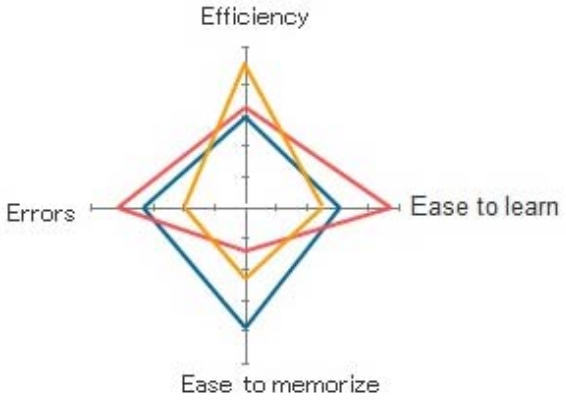

Figure 4. An example of usability evaluation result

To develop software with high usability, it is much important for developers and stakeholders to decide usability target as quantitatively and agree with each other. To achieve this, we have developed quantitative usability evaluation method using usability checklist [8]. That is not influenced by an evaluator’s subjective impression and preparing clear and precise definitions for checklist-based evaluations which are not affected by differences among evaluators. Results of evaluation were analyzed from four points of view (efficiency, errors, ease of learn and ease of memorize). For each item, weight and coefficient were set for analyzing. By showing the result using radar chart, we can understand which item is necessary to improve usability. Fig. 4 shows an example of evaluation result.

In ISO TC159 (Ergonomics) SC4 (HCI), we have tackled embodiment of human centered design process (HCDP), that is, this standard describes what shall each level in organization (executive, management, expert) do to apply human HCDP to development process [9]. From a viewpoint of quality, common industrial format for usability (CIF) were published [10-12]. They are “context of use description”, “user needs report” and “evaluation report”. About “user requirements specification” are discussing. These standards are related to software engineering. However both HCDP standards and CIF standards were not combined to software development process.

B. Cognitive scientific approach

Seffah [13, 14] and Ferre [6] pointed to that technical term, concepts, perception of the role, perspective and emphasis points in system and software development are different between software engineers and usability engineers who assume a role of usability improvement in system and software development project. They also pointed to that these differences caused miscommunication when software engineers and usability engineers collaborate in a system development project [13, 14] as well as disuse of usability improvement techniques by software engineers [6]. However they have not investigated specific cognitive features of software engineers.

We focus on the cognitive features of software engineers and investigate especially how software engineers understand the target system/software in the context of the development. Based on these features clarified, we consider why software engineers fail in design activities concerning usability [15].

From this, we found that tasks/work flow from system function and from the view points of user practice are different.

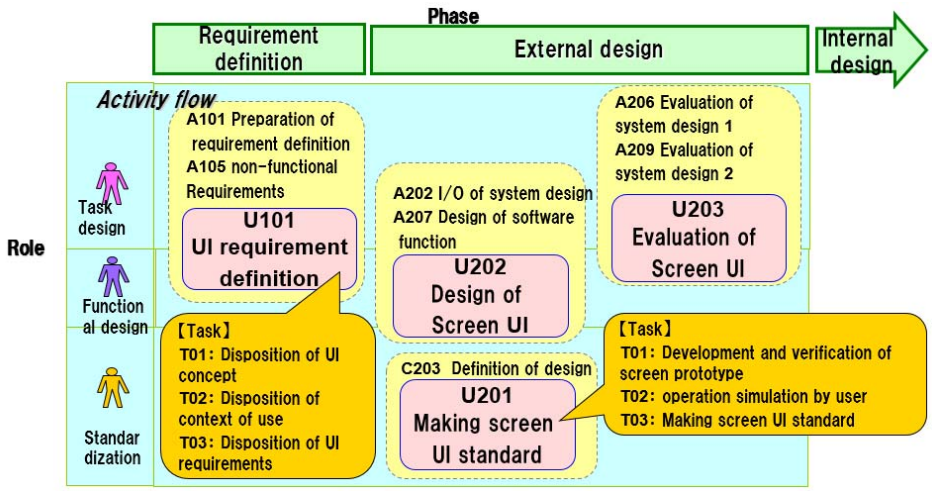

Figure 5. Development process combined with HCD activities

C. *Software engineering approach*

As shown in section 1, there are many UI guidelines for software development to make high usability software. However, almost guidelines do not indicate how to apply the guideline to development process and where in development process. To solve this, we made a UI design guide corresponding to HCDP decided in ISO9241-210 [1] and combined this guide with SI development process standard defined in our company. According to this process, SIers who do not have any knowledge about HCD or usability could apply HCDP to their development [16]. Development process combined with HCD activities is shown in Fig. 5 and correspondence HCD process and UI design procedure in development process is shown in Tab. 1. However, though this method is only a manual, sometimes SIers skip processes related to usability.

Table 1. Correspondence HCD process and UI design procedure in development process

UI Design Procedure	HCD Process
1. Defining the UI requirements - Organizing UI concept - Organizing the context of use	Understand and specify the context of use
- Organizing UI requirements	Specify the user and organizational requirements
2. Screen UI design creation - Develop and evaluate screen prototype - Operational simulation by users - Screen UI standard creation	(Small scale repetition of design work and user evaluation)
3. Screen UI design - Individual screen UI design	Produce design solutions
4. Screen UI evaluation - Screen UI standard conformity check - UI requirements list conformity check - Usability evaluation	Evaluate designs against requirements

III. FUTURE TARGETS

Generally, software, system and service are mainly developed based on software engineering. Definition of requirements specification phase were prescribed as development process. It is easy to evaluate whether developed system satisfies their requirements or not because the relationship between each phase of development, namely requirements specification, external design, internal design and function development, and each corresponding test phase is clear [17].

In definition of requirements specification phase, functional requirements and non-functional requirements shall be defined. Functions are that software operation which transforms input to output, and it is easy to evaluate or judge whether functional requirements can be realized or not. However, it is relatively difficult to define non-functional requirements and quality requirements and determine requirement level. So, it is also difficult to evaluate or judge whether these requirements can be realized or not.

In software development, it is difficult to define a non-functional requirement which descries “extent which provides feature of functions to users suitably”, because generally development processes prescribe only the things after the requirement definition phase.

On the other hand, we found that the difference of view points about software development between software engineers and usability experts shown in section 2.

Toward the future, we propose three issues to realize software development process with high usability.

1) Clarification of “demand” and evaluation process of “demand”, “user needs” and “user requirements” shown in Fig. 6 [18].

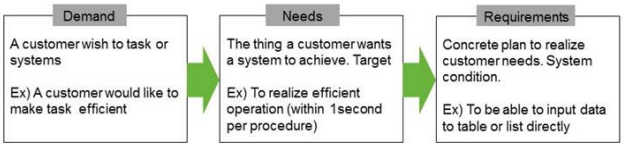

Figure 6. Relationship among “demand”, “needs” and “requirements”

2) Development of methods to fill gaps between software engineers and usability experts. They may be not only “manual” but also “tools” and “education”

3) Solution of twin peaks problem (gap between requirements, derived from users viewpoint, and software architecture decided from the viewpoint of structure). Even if an iterative or spiral process model would be applied, there still would be a gap between prioitization of development from each view point.

Especially, it is important to consider “quality”. CIF standards described in section II A. are parts of quality standards. This standard series include quality model, quality measurement and quality requirement. As there are some reports about quality model [19], for these issues, it is necessary to discuss detail targets.

REFERENCES

[1] ISO9241-210: Ergonomics of human-system interaction –Part210: Human-centred design for interactive systems (2010).

[2] Tanikawa, Y., Suzuki, H., Kato, H., Fukuzumi, S. : Problems in Usability Improvement Activity by Software Engineers - Consideration through verification experiments for human-centered design process support environment; HCI International 2014 Proceedings Vol.12 LNCS8521, pp.641-651, 2014.

[3] Roto, Virpi., Law, Effie., Vermeeren, Arnold. And Hoonhout, Jettie (ed.): Use Experience White Paper - Bringing clarity to the concept of user experience, Result from Dagstuhl Seminar on Demarcating User Experience, September 15-18, 2010, pp1-12, 2010.

[4] ISO/IEC 25062: Software engineering.— Software product Quality Requirements and Evaluation (SQuaRE) .— Common Industry Format (CIF) for usability Test Reports, (2005).

[5] Seffah, A., and Metzker, E.: The obstacles and myths of usability and software engineering;Communications of the ACM, 47(12), pp.71-76, 2004

[6] Ferre, X. : Integration of Usability Techniques into the Software Development Process; Proceedings of the ICSE Workshop on Bridging the Gaps between Software Engineeringand Human–Computer Interaction, pp.28-35, 2003

[7] Tanikawa, Y., et.al. : Problems in Usability Improvement Activity by Software Engineers -Consideration through verification experiments for human-centered design process supportenvironment; HCI International 2014 Proceedings Vol.12 LNCS8521, pp.641-651, 2014.

[8] Fukuzumi, S., et.al. : Development of Quantitative Usability Evaluation Method. of HCI International 2009 Proceedings, pp.252-258, 2009

[9] ISO DIS9241-210: Ergonomics of human-computer interaction -Part 220: Processes for enabling, executing and assessing human centred design within organizations (2017).

[10] ISO/IEC 25063: Software engineering. -Software product Quality Requirements and Evaluation (SQuaRE). - Common Industry Format (CIF) for usability: Context of use description (2014).

[11] ISO/IEC 25064: Software engineering. -Software product Quality Requirements and Evaluation (SQuaRE). - Common Industry Format (CIF) for usability: User needs report (2015)

[12] ISO/IEC 25066: Software engineering. -Software product Quality Requirements and Evaluation (SQuaRE). - Common Industry Format (CIF) for usability: Evaluation report (2016)

[13] Seffah, A., and Metzker, E.: The obstacles and myths of usability and software engineering; Communications of the ACM, 47(12), pp.71-76, 2004

[14] Seffah, A., Gulliksen, J., & Desmarais, M. C. (Eds.). : Human-Centered Software Engineering-Integrating Usability in the Software Development Lifecycle (Vol. 8). Springer Science & Business Media, 2005

[15] Tanikawa, Y., et al.: Modeling how to understand a target system: bridging the gap between software engineers and usability experts, HCII2016 Proceedings, pp. 220–232, 2016

[16] Hiramatsu, et.al. : Applying Human-Centered Design Process to SystemDirector Enterprise Development Methodology. NEC Technical Journal Volume vol.3 no.2, pp.12-16, 2008.

[17] V-Modell XT, http://www.v-modell-xt.de, 2005

[18] Shin’ichi Fukuzumi and Yukiko Tanikawa: Clarification of customers’ “demand” in development process, HCII2016 proceedings, pp. 413–420, 2016

[19] Richard Berntsson Svensson, Thomas Olsson, and Björn Regnell: Introducing Support for Release Planning of Quality Requirements - An Industrial Evaluation of the QUPER Model, Second InternationalWorkshop on Software Product Management (IWSPM-’08), Barcelona, Spain, 2008.