Aspects of Integrating User Centered Design into Software Engineering Processes

Karsten Nebe[1], Dirk Zimmermann[2]

[1] University of Paderborn, C-LAB, 33098 Paderborn, Germany
[2] T-Mobile Deutschland GmbH, Landgrabenweg 151, 53227 Bonn, Germany

[1] Karsten.Nebe@c-lab.de, [2] Dirk.Zimmermann@t-mobile.de

Abstract. Software Engineering (SE) and Usability Engineering (UE) both provide a wide range of elaborated process models to create software solutions. Today, many companies have realized the need for usable products and understood that a systematic and structured approach to usability is as important as the process of software development itself. However, theory and practice still have problems to efficiently and smoothly incorporate UE methods into established development processes. One challenge is to identify integration points between the two disciplines SE and UE that allow a close collaboration, with acceptable additional organizational and operational effort. The approach presented in this paper identifies integration points between software engineering and usability engineering on the level of process models. The authors analyzed four different software engineering process models to determine their ability to create usable products. Therefore, the authors synthesized demands of usability engineering and performed an assessment of the models.

Keywords: Software Engineering, Usability Engineering, Standards, Models, Processes, Integration, Assessment.

1 Introduction

The question how to align the two disciplines of Software and Usability Engineering originates from the need of usability practitioners to embed their activities into an existing organization that already applies methods for structured software development. Often a standalone usability process often does not have sufficient impact and its results are not readily absorbed by the development organization. Therefore it would be desirable to identify potential points for exchanging information or possibly sharing resources and efforts to achieve a situation where the benefits of usability engineering become usable for the development organization. The paper aims to identify these integration points on different levels and to create a framework for aligning the two disciplines.

1.1 Software Engineering

Software engineering is a discipline that adopts various engineering approaches to address all phases of software production, from the early stages of system specification up to the maintenance phase after the release of the system ([15], [18]). Software engineering tries to provide a systematic and planable approach for software development. To achieve this, it provides comprehensive, systematic and manageable procedures: so called software engineering process models (SE Models).

SE Models usually define detailed activities, the sequence in which these activities have to be performed and resulting deliverables. The goal of SE Models is to define a process where the project achievement does not depend on individual efforts of particular people or fortunate circumstances [5]. Hence, SE Models partially map to process properties and process elements and add concrete procedures. Existing SE Models vary with regards to specific properties (such as type and number of iterations, level of detail in the description or definition of procedures or activities, etc.) and each model has specific advantages and disadvantages, concerning predictability, risk management, coverage of complexity, generation of fast deliverables and outcomes, etc. Examples of such SE Models are the Linear Sequential Model (also called Classic Life Cycle Model or Waterfall Model) [16], Evolutionary Software Development [12], the Spiral Model by Boehm [1], or the V-Model [9].

Software engineering standards define a framework for SE Models on a higher abstraction level. They define rules and guidelines as well as properties of process elements as recommendations for the development of software. Thereby, standards support consistency, compatibility and exchangeability, and cover the improvement of quality and communication.

The ISO/IEC 12207 provides such a general process framework for the development and management of software [7]. It defines processes, activities and tasks and provides descriptions about how to perform these items on an abstract level.

Thus, there is a hierarchy of different levels of abstractions for software engineering: Standards that define the overarching framework and process models describe systematic and traceable approaches for the implementation. All these levels put the focus on system requirements and system design.

1.2 Usability Engineering

Usability Engineering is a discipline that is concerned with the question of how to design software that is easy to use. Usability engineering is “an approach to the development of software and systems which involves user participation from the outset and guarantees the efficacy of the product through the use of a usability specification and metrics.” [4]

Therefore usability engineering provides a wide range of methods and systematic approaches to support the development process. These approaches are called Usability Engineering Models (UE Models). Examples are Goal-Directed-Design [2], the Usability Engineering Lifecycle [11] or the User-Centered Design-Process Model of IBM [6]. They describe an idealized approach to ensure the development of usable

software, but they usually differ in its details, in the applied methods (the “how?”) and the general description of the procedure (the “what?”, e.g. phases, dependencies, goals, responsibilities, etc.) [19].

Usability engineering provides standards which are similar to the idea of software engineering standards. They also serve as a framework to ensure consistency, compatibility, exchangeability, and quality. However, usability engineering standards lay the focus on the users and the construction of usable solutions during the development of software solutions. Examples for such standards are the DIN EN ISO 13407 [3] and the ISO/PAS 18152 [8].

The DIN EN ISO 13407 introduces a process framework for the human-centered design of interactive systems. Its overarching aim is to support the definition and the management of human-centered design activities. The ISO/PAS 18152 is based on the DIN EN ISO 13407 and describes a reference model to measure the maturity of an organization in performing processes that make usable, healthy and safe systems.

Thus, in usability engineering exists a similar hierarchy of abstraction levels as in software engineering: Standards define the overarching framework and process models describe systematic and traceable approaches for the implementation. However, usability engineering puts the focus on creating usable and user-friendly systems instead of system requirements and system design.

1.3 Relationship of Standards, Models and Operational Processes

In general standards and models are seldom applied directly, neither in software engineering nor in usability engineering. Standards merely define a framework to ensure compatibility and consistency and to set quality standards. Models are being adapted and/or tailored according to the corresponding organizational conditions, such as existing processes, organizational or project goals and constraints, legal policies, etc. According to this, the models are detailed by the selection and definition of activities, tasks, methods, roles, deliverables, etc. as well as responsibilities and relationships in between. The derived instantiation of the model, fitted to the organizational aspects, is called software development process (for SE Models) or usability lifecycle (for UE Models). Thus, the resulting Operational Process is an instance of the underlying model and the implementation of activities and information processing within the organization. This applies to both software engineering and usability engineering.

Thus, there is not just a single hierarchy of standards and models but an additional level of operational processes for software engineering, as well as for usability engineering. Standards define the overarching framework, models describe systematic and traceable approaches and on the operational level these models are adjusted and put into practice (Figure 1).

In order to achieve sufficient alignment between the two disciplines, all three levels have to be regarded to ensure that the integration points and suggestions for optimized collaboration meet the objectives of both sides and not lose the intentions behind a standard, model or operational implementation.

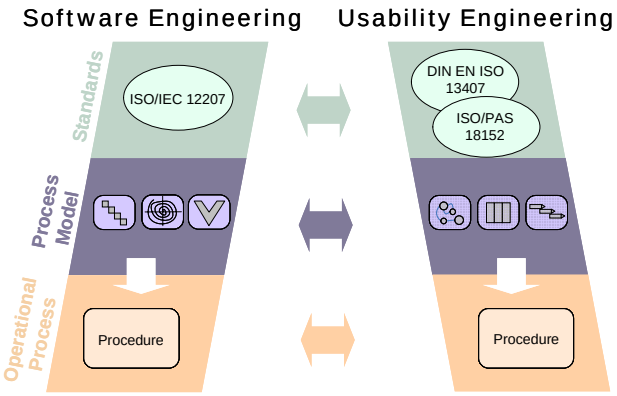

Fig. 1. Similar hierarchies in the two disciplines software engineering and usability engineering: standards, process models and operational processes.

2 Motivation

For development organizations SE Models are an instrument to plan and systematically structure the activities and tasks to be performed during software creation. However, software development organizations aim to fulfill specific goals when they plan a software solution. Such goals could be the rapid development of a new software solution (to become the leader in this area) or to develop a very stable and reliable solution (e.g. because of the organization's prestige) and of course, to create revenue with it. Depending on its' goals an organization will chose one (or the combination of more than one) SE Model for the implementation that will in their estimate fits best. As an example, the Linear Sequential Model with its predefined results at the end of each phase and its sequential flow of work certainly provides a good basis for planability. On the other hand, the Evolutionary Development might not be a good choice if the main focus of the solution is laid on error-robustness, because the continuous assembling of the solution is known to cause problems in structure and the maintenance of software code.

As usability engineering puts the focus on the user and usability of products, which is an important aspect of quality, usability becomes important for the development process and thus also an important criterion for organizations to choose a well-suited SE Model.

However, usability engineering activities are not just a subset of software engineering or SE activities. Although different models exist for software and usability engineering, there is a lack of systematic and structured integration [17]. They often coexist as two separate processes in an organization and therefore need to be managed separately and in addition need to be synchronized, by adding usability engineering activities to the software engineering process models.

In order to identify integration points between the two disciplines the authors believe examinations on each level of the hierarchy have to be performed: On the level of standards it has to be shown that aspects of software engineering and usability

engineering can coexist and can be integrated, even on this abstract level. On the level of process models it has to be analyzed how usability engineering aspects can be incorporated into SE Models. And on the operational level's activities, a close collaboration should be achieved, resulting in reasonable additional organizational and operational efforts.

2.1 Common Framework on the Level of Standards

In previous work the authors already performed an initial analysis on the first two hierarchy levels [13] of Standards and Processes. First integration points on the level of Standards could be found in comparing the software engineering standard ISO/IEC 12207 with the usability engineering standard DIN EN ISO 13407. Therefore, standards' detailed descriptions of processes, activities and tasks, output artifacts, etc. have been analyzed and similarities were found. Based on common goals and definitions, the single activities of the standards could be consolidated as five common activities: *Requirement Analysis*, *Software Specification*, *Software Design and Implementation*, *Software Validation* and *Evaluation*. These common activities represent and divide the process of development from both, a software engineering and a usability engineering point of view.

The five common activities can be seen as basis for integrating the two disciplines on the overarching level of standards: a common framework for software engineering and usability engineering activities. The authors used the framework to set the boundaries for the next level of analysis in the hierarchy: the level of process models.

2.2 Ability of SE Models to Create Usable Products

Based on the common framework different SE Models were analyzed with regards to see how they already support the implementation of the usability activities. Thus, an assessment of SE Models with the goal to identify the ability of SE Models to create usable software solutions was performed.

In order to create valuable results, the authors defined several tasks to be performed. First, adequate criteria for the assessment of the SE Models needed to be defined, by which unbiased and reliable statements about process models and their ability to create usable software can be made. The assumption was that based on the results of the assessment specific recommendations can be derived to enrich the SE Models by adding or adapting usability engineering activities, phases, artifacts, etc. By doing this, the development of usable software on the level of process models can be guaranteed. Furthermore, hypothesizes about the process improvements can be made for each recommendation which then can be evaluated on the Operational Process level. Therefore, case studies will be identified based on which the recommendations can be transferred in concrete measures. These measures can then be evaluated by field-testing to verify their efficiency of user-centeredness of software engineering activities.

In summary, four types of analyses need to be performed: two on the level of process models and two on the operational process level. The four respective analysis topics differ in their proceedings as well as their expected results:

- Operationalization of the base practices and the identification of criteria for the assessment von usability engineering activities and the corresponding deliverables.
- Assessment of SE Models, based on the identified criteria and the derivation of adequate recommendations.
- Inspection of case studies with regards to the recommendations and the derivation of specific measures for the implementation of UE activities in SE Processes
- Evaluation of the measures in practice

For each of the analyses several methods can be used, some of which involve domain experts as interview partners, whereas others are more document oriented. This paper focuses on the description of the performed analyses in the first topic listed above and first results on the second topic as a forecast based on the results of the first topic, i.e. the operationalization of base practices and derivation of UE criteria for the assessment.

3.1 Criteria for the Assessment of SE

As the authors identified the need for assessment criteria to define the degree of usability engineering coverage in SE Models, the following section shows how these criteria were gathered and what results were derived and to be expected from further research activity.

To obtain detailed knowledge about usability engineering activities, methods, deliverables and their regarding quality aspects, the authors analyzed the DIN EN ISO 13407 and the ISO/PAS 18152. In addition to the identified common activities of the framework within the human-centered design activities, ISO/PAS 18152 defines detailed Base Practices that specify the tasks for creating usable products. These base practices have been used as a foundation to derive requirements that represent the common activities' usability engineering perspective. The quantity of fulfilled requirements for each activity of the framework informs about the level of compliance of the SE Model satisfying the base practices and therewith the usability view of activities. For each base practice the authors determined whether the model complied with it or not.

In a second iteration of the gap-analysis expert interviews will lead to more detailed criteria in order to assess the corresponding SE Models more specific. Additionally the completeness and correctness of the base practices and human-centered design activities as defined in the ISO/PAS 18152 itself needs to be verified.

The detailed descriptions of the base practices have been used to pre-structure the collection of criteria and for the expected results. Since the base practices are structured based on activities, methods, and deliverables the authors used this to pre-structure the expected results. Additionally expected results are criteria about the quality aspects of the overall process. The results will be separated based on the

specific human-centered design activities and those that are more generic and overarching. This results in a matrix of activities & methods, content & deliverables, roles & quality aspects in relation to the human-centered design and overall activities as shown in Table 2.

Table 2. Structure and orientation of criteria for the assessment of software engineering models

	Overarching Aspects	Context of use	User Requirement	Produce Design	Evaluation of use
Activities & Methods					
Deliverables & Content					
Roles					
Quality Aspects					

Based on this, several evaluation questions have been gathered, focusing on the abstract level of process models. The goal is the to define overarching criteria and not evaluate the concrete accomplishment within one specific model or particular procedure, e.g. questions about overlaps of activities, phases, deliverables, or questions about the relevance of specific activities or roles within a process model.

According to the questions and based on the initial structure, as shown in Table 2, the authors performed the first analysis, the documentation of existing SE Models (Linear Sequential Model, Evolutionary Software Development, the Spiral Model by Boehm and the V-Model) and for the second analysis created an interview guideline that is currently used as basis for the expert-interviews. Initial results of theses analyses are described in the following section.

4 Results

As a result of the first analysis of selected SE Models first general statements can be made: The overall level of compliance of the SE Models satisfying the base practices and therewith the usability view of activities, is rather low. For none of the SE Models all base practices of ISO/PAS 18152 are fulfilled. However, there is also a large variability in the coverage rate between the SE Models. For example, the V-Model shows a very good coverage for all modules except for lower compliance of the activity HS 3.3 *Produce Design Solution* criteria, whereas the Linear Sequential Model only fulfills a few of the HS 3.4 *Evaluation of Use* criteria and none of the other modules. Evolutionary Design and the Spiral Model share a similar pattern of findings, in that they show little coverage for *Context of Use*, medium to good coverage of *User Requirements*, limited coverage for *Produce Design Solution* and good support for *Evaluation of Use* activities.

Table 3. Summary Results of the gap-analysis, showing the sufficiency of SE Models in covering the requirements of usability engineering (based on the ISO/PAS 18152; HS 3).

	Context of Use	**User Requirements**	**Produce Design Solutions**	**Evaluation of Use**	***Across Activities***
Linear Sequential Model	**0 %**	**0 %**	**0 %**	**60 %**	***13 %***
Evolutionary Development	**13 %**	**40 %**	**40 %**	**80 %**	***39 %***
Spiral Model	**13 %**	**80%**	**40 %**	**100 %**	***52 %***
V-Modell	**88 %**	**80 %**	**40 %**	**100 %**	***78 %***
Across Models	***28 %***	***50 %***	***30 %***	***85 %***	

By looking at the summary of results and comparing the percentage of fulfilled requirements for each SE Model, it shows that the V-Model has a better compliance than the other models and it can basically be regarded to be able to produce usable products. In the comparison, the Linear Sequential Model cuts short, followed by Evolutionary Development and the Spiral Model.

Both in the overview and the detail findings it shows that the emphasis for all SE Models is laid on evaluation (*Evaluation of Use*), especially in comparison to the remaining activities. The lowest overall coverage could be found in Context of Use and Produce Design Solution.

Based on the relatively small compliance values for the *Context of Use* (28%), *User Requirements* (50%) and *Produce Design Solutions* (30%) activities across all SE Models, the authors see this as an indicator that there is only a loose integration between usability engineering and software engineering.

In summary, the results confirmed expectations of the authors, showing the low level of integration between both disciplines on the level of the overarching process models. As expected it becomes apparent that there is a dire need to compile more specific and detailed criteria for the assessment of the SE Models. As the analysis showed, the base practices currently give too much leeway for interpretations.

In addition it turned out that the dichotomous assessment scale (in terms of “not fulfilled” or “fulfilled”) is not sufficient. A less granular rating is necessary to evaluate the process models adequately.

Performing the documentation analysis of the SE Models produced first insights but it turned out that the documentation is not comprehensive enough to ensure the validity of the resulting statements.

In the second analysis the authors plan to conduct more specific criteria will be determined, according to the previously described structure. These will be compiled in semi-structured interviews with experts from the domain of usability engineering.

The criteria focus on the activities defined in the module Human-centered design (ISO/PAS 18152) and their respective base practices and specifics in: fundamental

activities, basic conditions and constraints, relevance of activities, resulting outcomes, type of documentation, and respective roles and responsibilities. Beyond this, a substantial focus is put on the quality aspects based on the activities, deliverables, roles and the super ordinate model. The criteria will be evaluated concerning questions like:

- How to identify good activities?
- How to identify good results or deliverables?
- How to identify appropriate Roles
- What are properties/characteristics for the relevance and frequency?
- How could the progress of an activity or deliverable be measured and controlled?

Based on these criteria the authors expect to be able to get evidence, which activities, deliverables and roles are necessary to ensure the development of usable products from the experts' point of view. Relevant factors of influence could be for instance: „When will an activity A not be performed, and why?" or "Under which circumstances will an activity A be performed completely, when just partly?" Additionally, criteria are to be raised, based on which the progress of the process could be measured.

However, the central point will be collection of criteria that focus on quality aspects of the activities, deliverables and roles as well as their relevance.
It is expected that the results can not just be used as more detailed criteria for the assessment but will also provide evidence on the level of completeness of the ISO/PAS 18152 and surface potential areas of improvement.

5 Summary & Outlook

The approach presented in this paper was used to identify integration points between software engineering and usability engineering on the level of process models. The authors analyzed four different software engineering process models to identify their ability to create usable products. The authors synthesized demands of usability engineering and performed an assessment of the models.

The results provide an overview about the degree of compliance of the models with usability engineering demands. It turned out that there is a relatively small compliance to the usability engineering activities across all software engineering models. This is an indicator that there only little integration between usability engineering and software engineering exists. There are less overlaps between the disciplines regarding these activities and therefore it is necessary to provide suitable interfaces to create a foundation for the integration.

The authors identified the need to compile more specific and detailed criteria for the assessment as well as a more differentiated dichotomous assessment scale to evaluate the process models appropriately. Therefore the authors introduced a structured approach of how they will perform the follow-up analysis. The more detailed criteria will be compiled in semi-structured interviews with experts from the domain of usability engineering. Thereby, a substantial focus is put on the quality aspects based on the activities, deliverables, roles and the super ordinate model. Based

on these criteria the authors expect to be able to make statements about their necessity and the relevance to ensure the development of usable products from the experts' point of view. It is expected that the results could not just be used as criteria for the assessment of software engineering models but could also define the demands of usability more precisely and to give evidence about the completeness and potential extension areas of the ISO/PAS 18152.

References

1. Boehm, B.: A Spiral Model of Software Development and Enhancement. IEEE Computer. Vol. 21 (1988) 61-72
2. Cooper, A. & Reimann, R.: About Face 2.0. Wiley, Indianapolis, IN (2003)
3. DIN EN ISO 13407. Human-centered design processes for interactive systems. CEN - European Committee for Standardization, Brussels (1999)
4. Faulkner, X.: Usability Engineering. PALGARVE, New York (2000) 10-12
5. Glinz, M.: Eine geführte Tour durch die Landschaft der Software-Prozesse und – Prozessverbesserung. Informatik – Informatique. (6/1999) 7-15
6. IBM: Ease of Use Model. Retrieved from http://www-3.ibm.com/ibm/easy/eou_ext.nsf/publish/1996, (11/2004)
7. ISO/IEC 12207. Information technology - Software life cycle processes. Amendment 1, 2002-05-01. ISO copyright office, Switzerland (2002)
8. ISO/PAS 18152. Ergonomics of human-system interaction — Specification for the process assessment of human-system issues. First Edition 2003-10-01. ISO copyright office, Switzerland (2003)
9. KBST: V-Modell 97. Retrieved from www.kbst.bund.de (05/2006)
10. Larman, C., Basili, V.R.: Iterative and Incremental Development: A Brief History. Computer. Vol. 36. no. 6. (6/2003) 47-56
11. Mayhew, D. J.: The Usablility Engineering Lifecycle. Morgan Kaufmann, San Francisco (1999)
12. McCracken, D.D., Jackson M.A.: Life-Cycle Concept Considered Harm-ful. ACM Software Engineering Notes (4/1982) 29-32
13. Nebe K., Zimmermann D.: Suitability of Software Engineering Models for the Production of Usable Software. Proceedings of the Engineering Interactive Systems 2007, HCSE (IFIP Working Group 13.2, Methodologies for User Centered Systems Design), Lecture Notes In Computer Science (LNCS). Springer (2007) in prep.
14. Pagel, B., Six, H.: Software Engineering: Die Phasen der Softwareentwicklung. Vol. 1. 1st ed. Addison-Wesley Publishing Company, Bonn, D (1994)
15. Patel, D., Wang, Y (eds.): Annals of Software Engineering. Editors' introduction: Comparative software engineering: Review and perspectives. Vol. 10. Springer, Netherlands (11/2000) 1-10
16. Royce, W. W.: Managing the Delopment of Large Software Systems. Proceedings IEEE. IEEE, Wescon (1970) 328-338
17. Seffah, A. (ed.): Human-Centered Software Engineering – Integrating Usability in the Development Process. Springer, Dordrecht, The Netherlands (2005) 3-14
18. Sommerville, I.: Software Engineering. 7th ed. Pearson Education Limited, Essex, GB (2004)
19. Woletz, N.: Evaluation eines User-Centred Design-Prozessassessments - Empirische Untersuchung der Qualität und Gebrauchstauglichkeit im praktischen Einsatz. Doctoral Thesis. University of Paderborn, Paderborn, Germany (4/2006)