Usability-Engineering-Requirements as a Basis for the Integration with Software Engineering

Karsten Nebe¹ and Volker Paelke²

¹ University of Paderborn, C-LAB, Fürstenallee 11, 33098 Paderborn, Germany ² Leibniz University Hannover, Appelstrasse 9a, 30167 Hannover, Germany Karsten.Nebe@c-lab.de, Volker.Paelke@ikg.uni-hannover.de

Abstract. Usability is growing to become an integral quality aspect of software development, but it is not an exclusive attribute of the generated product; it is also a fundamental attribute for the development process itself. The question is how to adapt software engineering processes (or models) in such a way that they can ensure the development of usable solutions. In this paper, the authors present an integration approach pursuing this goal. It draws on so called 'Compliancy and Key Requirements' that can be used for the definition of software processes (or process models) and thereby support the integration of both disciplines. The requirements are based upon representative standards (DIN ISO 13407 and ISO/PAS 18152) but were enhanced by the results of an expert based survey using interviews and questionnaires. Additionally the requirements have been verified by experts and represent an evaluated knowledge base for the development of usable products.

Keywords: Integration, Software Engineering, Usability Engineering, Standards DIN EN ISO 13407 and ISO/PAS 18152, Process Models, Process Definition, Process Improvement, Assessment.

1 Introduction

The integration of Software Engineering (SE) and Usability Engineering (UE) is a widespread topic in theory and practice as the need for usability grew to prominence in software development within the last years. There are far-reaching benefits both for the users, which include a high productivity, increased quality of work and user-satisfaction, as well as for the organizations, e.g. in monetary form, such as the reduction of support- and training-costs [10]. Thus, the usability of the solution became an integral quality aspect in software development, especially in comparison with competing products (respectively competing manufacturers) and can result in a unique selling proposition.

But quality is not an exclusive attribute of the generated product; it is also a fundamental attribute of the manufacturing process itself. An optimal process would be designed in such a way as to assure the desired quality of the produced solution. At this point UE methods are used to ensure utilizable solutions. However, in industrial practice UE can only exist in addition to or in combination with SE. Hence, there is a need to integrate the two disciplines of SE and UE. The goal is to combine the procedures and the goals of SE and UE in a way that allows systematic and predictable implementations to be generated while considering the factors of costs, time and quality adequately for both SE and UE proposes.

In this paper, the authors present an integration approach pursuing this goal.

2 Integration Approaches

In theory and praxis, a considerable number of integration approaches with distinct focuses exist [18]. Some of these approaches tend to define common activities and artifacts for both SE and UE and to integrate these specific activities into the process of development. They aim at a 'soft integration' of UE aspects on a mutual basis, e.g. at interlinking relative results (e.g. [17, 5, 2]). Most approaches focus on a minimal organizational and structural transformation and/or change. Quite similar are approaches that aim at a common specification of activities and artifacts. They are grounded on communication and information exchange by using shared definitions (e.g. [1, 21, 20]). These two kinds of approaches could be summarized as a group of approaches that aim directly at the operational development processes in organizations.

Other integration approaches relate to the level of process definitions and process models (e.g. [6, 11,3]). These aim to define pre-settings for the development and contain both a more concrete approach (focusing on the integration of UE activities in an already existing SE-Models), and more fundamental aspects of process models (independently of any concrete SE-Model). In general these approaches concentrate on the combination of phases, activities and results (within existing structures) on the level of process models to build up the base for integration.

In addition there is another third group of integration approaches focusing on a higher level ob abstraction. Those approaches are independent from any specific process model or activities but rather describe organizational measures, principles, paradigms or meta-models (e.g. [16, 7, 5, 19]). Those approaches aim at the definition of general procedures for the development, which is comparable to standards in SE and UE on this level of abstraction. Accordingly, strategies for the implementation are abstract and need to be adapted to particular situations.

Altogether these groups of approaches aim to provide systematic procedures for developing usable software. At a closer look, they address three different levels of abstraction:

- 1. The abstract overarching level of standards in software engineering and usability engineering, serving as a framework to ensure consistency, compatibility, exchangeability, and quality within and beyond the organizational borders and to cover the improvement of quality and communication.
- 2. The level of process models for software engineering and usability engineering, to provide a procedural model that can serve as a framework for an organization, providing specific features, e.g. predictability, risk management, coverage of complexity, generation of fast deliverables and outcomes, etc.
- 3. The operational process level which reflects the execution of activities and the processing of information within the organization. It is an instance of the underlying model and the implementation of activities and information processing within the organization.

These are related in a hierarchy: standards define the overarching framework, process models describe systematic and traceable approaches within such a framework, and at the operational level the models are tailored to fit the specifics of an organization.

2.1 Integration on the Level of Standards, Process Models and Operational Process

It can be observed that this hierarchy of standards, process models and processes exists in both disciplines, but there have been few attempts to exploit these similarities for integration. With this goal in mind, the authors analyzed these three levels and presented a holistic approach for integration of SE and UE [12, 13, 14].

By doing this, the authors identified similarities between SE and UE on the level of standards. The standards' detailed descriptions of processes, activities and tasks, output artifacts, etc. were analyzed and compared. For this, the SE standard ISO/IEC 12207 [8] was chosen for comparison with the UE standard DIN EN ISO 13407 [4]. On a high level, when examining the descriptions of each activity, by relating tasks and outputs with each other, similarities could be identified in terms of the characteristics, objectives and proceedings of activities. Based on these similarities single activities were consolidated as groups of activities (so called, 'common activities'). These 'common activities' are part of both disciplines SE and UE on the highest level of standards. The result is a compilation of five 'common activities': Requirement Analysis, Software Specification, Software Design and Implementation, Software Validation, Evaluation that represent the process of development from both, a SE and a UE point of view [12, 13]. These activities define the overarching framework for the next level, the 'level of process models'.

In a following analysis the authors identified the maturity of software engineering process models' ability to create usable products [12, 14]. For that purpose, the authors used a two-step approach to synthesize the demands of usability engineering and performed an assessment of selected software engineering models.

To obtain detailed knowledge about usability engineering activities, methods, deliverables and their regarding quality aspects, the authors analyzed the two usability engineering standards DIN EN ISO 13407 and the ISO/PAS 18152 [9]. The ISO/PAS 18152 defines detailed base practices that specify the tasks for creating usable products. These base practices were used as a foundation to derive requirements that represent the 'common activities' from a usability engineering perspective. The quantity of fulfilled requirements for each activity of the framework informs about the level of compliance of the software engineering model. It provides an estimate of how well the UE base practices are covered in a given SE model.

The results of the assessment provide an overview about the degree of compliance of the selected models with usability engineering demands. It turned out that there is a relatively small compliance to the usability engineering activities across all selected software engineering models. This is an indicator that only little integration between usability engineering and software engineering currently exists on the level of process models.

The analysis did not only highlight weaknesses of SE Models, it also pinpointed the potential for integration between software engineering and usability engineering: Where base practices are not considered as fulfilled, recommendations could be derived, which would contribute to an accomplishment. The underlying base practices provide indices what needs to be considered on the level of process models. This can be used a foundation for implementing the operational process level.

However, during the analysis it became apparent that there is a clear need for more detailed/adequate criteria for the assessment by which more objective and reliable statements about process models and their ability to create usable software could be made.

Such detailed criteria would also be useful to formalize process-requirements that can influence the definition of SE-Models and development processes that are usercentered and by improve the interplay of SE and UE in practice.

Having this in mind, the authors performed semi-structured interviews with experts from the domain of UE to identify requirements from the UE perspective. The results have been analyzed and evaluated as described in the following paragraph.

3 UE-Process-Requirements

In order to make software development processes user-centered there is a need for explicit knowledge about relevant activities, their dependencies, regarding results, roles, and quality aspects, etc. One goal is to develop such a knowledge-base using existing findings and to enrich them by expert's knowledge.

Therefore the authors created an interview-guideline and questionnaires that correspond to the overall process framework of common activities particularly with regards to the usability engineering perspective. The analysis is based on the four humancentered design activities of the DIN EN ISO 13407 ('context of use', 'user requirements', 'produce design solutions' and 'evaluation of use') and their respective base practices and specifics as defined in the ISO/PAS 18152 (i.e. fundamental activities, basic conditions and constraints, relevance of activities, resulting outcomes, type of documentation, and respective roles and responsibilities). The goal was not to evaluate these standards but to add details for further use.

A substantial part of the analysis referred explicitly to quality characteristics of the four human-centered design activities. The goal was to identify what constitutes the quality of a certain activity from the experts' point of view and what kind of (potentially measurable) success and quality criteria exist that are relevant on a process level and subsequently for the implementation in practice. Examples of the questionnaire are: 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 the results the authors identified activities, deliverables and roles that 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 that allow measuring the progress of the development process.

It was expected that the results could be used not just as more detailed criteria for an assessment but would also provide an indication of the level of completeness of the ISO/PAS 18152 and identify potential areas of improvement.

To achieve this, the authors performed semi-structured interviews and questionnaires with six experts in the field of UE [15]. The experts were well grounded in theoretical terms, i.e. standards and process models, as well as in usability practice.

3.1 Derivation of Requirements

As a result, about 470 statements from the experts have been gathered which then have been consolidated and classified by adding references to its source (i.e. the interview partner and the question out of the interview-guideline); to one of the four activities ('context of use', 'user requirements', 'produce design solutions' or 'evaluation of use'); whether it addresses quality aspects regarding the process, an activity, or deliverable; whether it complies to the activities' and base practices' goals (as defined in the two ISO standards), etc. Thus, overarching process- and quality characteristics could be identified that led to findings about the relevance, the applicability and need of usability activities, methods and artifacts to be implemented in SE.

By performing several iterations of analysis similar statements were merged and formalized in terms of 107 'requirements for development processes or process models. There are two distinct types of requirements: 'Compliancy and Key Requirements'. Compliancy requirements represent the goals and base practices defined in the standards DIN EN ISO 13407 and ISO/PAS 18152 but refine them with the output of the analysis. The key requirements define core characteristics of the overall frameworks usability activities focusing on the activities' and results' quality. Together, the requirements define the demands of UE and lead to the systematically creation of usable products. Examples of the resulting requirements are:

- Context-analysis is an integral part of the process.
- Analysis takes part early in the process before conceptual work is carried out.
- Analysis activities are preformed iteratively until all incompletions and inconsistencies are eliminated.
- Resources and time for the elicitation and evaluation of user requirements is sufficiently provided.
- User requirements are addressed in the system design.
- User requirements are the input for the next process step and accordingly positioned in the development process.
- The requirements of the users of the system are defined.

3.2 Evaluation of Requirements

In a subsequent analysis, both the compliancy and key requirements have been evaluated by 13 usability experts using questionnaires (three of these experts were also involved in the previous analysis). The questionnaire included a list of all 107 requirements grouped by the four activities ('context of use', 'user requirements', 'produce design solutions' or 'evaluation of use') and scales to rate the correctness and the relevance for the appliance in practice of each requirement. Some examples of Requirements are shown in Table 1.

Table 1. Examples of the requirements for the UE-activites 'context of use' (CoU), 'user requirements' (UR), 'produce design solutions' (PDS) and 'evaluation of use' (EoU) and the experts' rating in terms of correctness and relevance (in practice)

Nr	Activity	Requirement	Correctness	Relevance
2	CoU	Context-analysis is an integral part of the process.	Correct	Very high
17	CoU	The outcomes of the context analysis serve as the	Correct	High
		input for the next process step and the activity itself is		
		anchored within the process model accordingly.		
27	CoU	The characteristics of the intended users and their	Correct	Very High
		tasks, including user interaction with other users		
		and other systems, are documented.		
24	CoU	The analysis is focused on the original context of the	Correct	High
		users (their goals, tasks, characteristics of the tasks		
		and the environment, etc.). The analysis is independ-		
		ent of any existing solution/implementation.		
33	CoU	The context-information is based on facts and not an	Sufficient	Medium
		interpretation of any situation.		
46	UR	A sufficient amount of user requirements are the basis	Correct	Very High
		for the next process step (PDS).		
71	PDS	The development of solutions is carried out in col-	Correct	Very High
		laboration with the development team.		
105	EoU	It is checked that the system is ready for evaluation.	Sufficient	Medium

By looking at the overall results it turned out that most requirements are rated correct by the majority of experts: 31 requirements by all 13 participants; 29 requirements by 12 experts; 27 requirements by 11; and 6 requirements by at least 10 Experts. No requirement has been rated incorrect. All together there is a high compliance of the experts opinions to the requirements. The sum of requirements that has been rated correct by at least 10 experts is 93 – which represent 87% of all 107 requirements.

The rating of the relevance was used to derive recommendations about the priority for the appliance in practice (i.e. for the definition of processes).

- 1. Those requirements that have been rated as 'correct' and range from a 'very high' to a 'high' scale of relevance.
 - (in general: the higher the relevance the higher the priority).
- 2. Those requirements that have been rated as 'correct' and show 'medium' scale of correctness.
- 3. Those requirements that depict a 'sufficient' scale of correctness.
- 4. Those requirements that show an 'acceptable' scale of correctness.
- 5. All remaining requirements.

But, by applying the requirements in practice, it is important to consider requirements of all four activities in equal measure. A partially implementation of selective requirements will not lead to usable products. Only using them in a holistic way will support the systematic development of usable solutions.

As a result of the analysis and evaluation the compliancy and key requirements represent an evaluated knowledge basis for the development of usable products. The analysis based on representative standards of UE and the requirements add here to more specific criteria based on experts' knowledge. The requirements account for the integration of SE and UE as they can be used for the definition and adaption of SE process models as well as operational development processes.

4 Conclusions and Outlook

In summary, there exist many integration approaches that aim to provide systematic procedures for developing usable software. At a closer look, they address three different levels of abstraction: standards, process models and operational processes. However, there have been few attempts to exploit the integration in a holistic way including all three levels. The authors report about such an approach and present a systematic way of integrating usability engineering demands into the software engineering methodology. The results of an expert based analysis (and subsequent evaluation) have been used to derive two distinct types of requirements: 'Compliancy and Key Requirements'. Compliancy requirements represent the goals and base practices defined in the standards DIN EN ISO 13407 and ISO/PAS 18152 but those are refined by the output of the analysis. The key requirements define core characteristics of the overall frameworks usability activities focusing on the activities' and results' quality and are also based of the analysis' results.

The requirements represent an evaluated knowledge basis for the development of usable products. They add to an integration of software engineering and usability engineering as they can be used for the definition and adaption of software development processes and process models, too.

In future we aim to evaluate these requirements in practical projects to observe process changes and their resulting effects to the usability of the products.

References

- 1. Constantine, L.L., Lockwood, L.A.D.: Software for Use: A Practical Guide to the Models and Methods of Usage-Centered Design. Addison-Wesley (ACM Press), New York (1999)
- Constantine, L.L., Biddle, R., Noble, J.: Usage-centered design and software engineering. Models for integration. In: IFIP Working Group 2.7/13.4, ICSE 2003 Workshop on Bridging the Gap Between Software Engineering and Human-Computer Interaction, Portland (2003)
- Düchting, M., Zimmermann, D., Nebe, K.: Incorporating User Centered Requirement Engineering into Agile Software Development. In: Jacko, J.A. (ed.) HCI 2007. LNCS, vol. 4550, pp. 58–67. Springer, Heidelberg (2007)
- 4. DIN EN ISO 13407: Human-centered design processes for interactive systems, CEN -European Committee for Standardization, Brussels (1999)
- Ferre, X.: Integration of Usability Techniques into Software Development Process. Bridging The Gaps Between Software Engineering and Human-Computer Interaction. In: Proceedings of ICSE 2003 International Conference on Software Engineering, pp. 28–35. ACM Press, Portland (2003)
- Göransson, B., Lif, M., Gulliksen, J.: Usability Design-Extending Rational Unified Process with a New Discipline. In: Jorge, J.A., Jardim Nunes, N., Falcão e Cunha, J. (eds.) DSV-IS 2003. LNCS, vol. 2844, pp. 316–330. Springer, Heidelberg (2003)

- Granollers, T., Lorès, J., Perdrix, F.: Usability Engineering Process Model. Integration with Software Engineering. In: Proceedings of the Tenth International Conference on Human-Computer Interaction, pp. 965–969. Lawrence Erlbaum Associates, New Jersey (2002)
- ISO/IEC 12207: Information technology Software life cycle processes, 2nd edn., 2008-02-01. ISO/IEC, Genf (2008)
- 9. ISO/PAS 18152: Ergonomics of human-system interaction Specification for the process assessment of human-system issues. ISO, Genf (2003)
- Jokela, T.: An Assessment Approach for User-Centred Design Processes. In: Proceedings of EuroSPI 2001, Limerick Institute of Technology Press, Limerick (2001)
- Kolski, C.: A call for answers around the proposition of an HCI-enriched model. ACM SIGSOFT Software Engineering Notes 23(3), 93–96 (1998)
- Nebe, K., Zimmermann, D.: Suitability of Software Engineering Models for the Production of Usable Software. In: Proceedings of the Engineering Interactive Systems 2007. LNCS, Springer, Heidelberg (2007a)
- Nebe, K., Zimmermann, D.: Aspects of Integrating User Centered Design into Software Engineering Processes. In: Jacko, J.A. (ed.) HCI 2007. LNCS, vol. 4550, pp. 194–203. Springer, Heidelberg (2007b)
- Nebe, K., Zimmermann, D., Paelke, V.: Integrating Software Engineering and Usability Engineering. In: Pinder, S. (ed.) Advances in Human-Computer Interaction, ch. 20, pp. 331–350. I-Tech Education and Publishing, Wien (2008b)
- Nebe, K.: Integration von Usability Engineering und Software Engineering: Konformitäts und Rahmenanforderungen zur Bewertung und Definition von Softwareentwicklungsprozessen, Doctoral Thesis, Shaker Verlag, Aachen (in print, 2009)
- Pawar, S.A.: A Common Software Development Framework For Coordinating Usability Engineering and Software Engineering Activities. Master Thesis, Blacksburg, Virginia (2004)
- 17. Schaffer, E.: Institutionalization of usability: a step-by-step guide. Addison-Wesley, Pearson Education, Inc., Boston (2004)
- Seffah, A., Desmarias, M.C., Metzker, E.: Human Centered Software Engineering, HCI, Usability and Software Engineering Integration: Present and Future. In: Seffah, A., Gulliksen, J., Desmarais, M.C. (eds.) Human-Centred Software Engineering – Integrating Usability in the Development Lifecycle, vol. 8, Springer, Heidelberg (2005)
- Sousa, K., Furtado, E., Mendoca, H.: UPi: a software development process aiming at usability, productivity and integration. In: Proceedings of the 2005 Latin American conference on Human-computer interaction CLIHC 2005, ACM Press, New York (2005)
- Tidwell, J.: Designing Interfaces: Patterns for Effective Interface Design. O'Reilly, Sebastopol (2005)
- 21. Van Harmelen, M.: Object Modeling and User Interface Design: Designing Interactive Systems. Addison-Wesley, Boston (2001)