Eliciting Requirements for a Adaptive Decision Support System through Structured User Interviews

Stephan Weibelzahl[1], Andreas Jedlitschka[2], and Brahim Ayari[2]

[1]National College of Ireland, Mayor Street, Dublin 1, Ireland
sweibelzahl@ncirl.ie
[2]Fraunhofer Institute Experimental Software Engineering, Fraunhofer-Platz 1, D-67663 Kaiserslautern, Germany
[jedl,ayari]@iese.fraunhofer.de

Abstract. Eliciting user requirements at an early stage of software development can safe development time and effort. However, identify requirements for adaptivity, such as inter-individual differences in needs or preferences is not trivial. In this paper we revisit results reported in a previous paper from a methodological point of view. Using an example, we argue that scenarios in combination with structured interviews are not able to adequately identify adaptivity requirements due to reasons inherent to the method, such as the users' trust and their ability to anticipate system funtionality. We suggest that more implicit methods must be used at early development phases to obtain unbiased results.

1 User-Centered Design and Adaptivity

User-adaptive systems are by definition interactive systems. While there is only little literature on guiding the development of user-adaptive systems from a software engineering point of view (e.g., [8]), user-centered design techniques seem to be an obvious process to be adopted for the design of this kind of systems. The rational of user-centered design (UCD) is to place the person as opposed to the software artifact at the center [7]. Users are involved in the development process in very early phases of the software development and in fact throughout the complete development life-cycle. Involving users from the very beginning can help to discover their mental models and expectations, to identify and analyze theory tasks, workflow and goals, and in general to validate the developers' assumptions about the users. As UCD focuses on cognitive factors (such as perception, memory, learning, problem-solving, etc.) it seems particularly suitable for user-adaptive systems.

In this paper we explore the suitability of one particular technique for eliciting user requirements: scenarios. We argue that scenarios used in combination with structured interviews are not able to adequately identify adaptivity requirements due to reasons inherent to the method. Using the example of the development of a decision support system, we show that scenarios did not inform the development process in the expected way. Reasons for this shortcoming are discussed.

2 Informing the Development of a Decision-Support System

In a previous paper [6] we outlined the development of a Decision-Support System in the domain of software engineering.

2.1 Software engineering decision support

Software engineering decision support (SE-DS) is an emerging field [9,10]. One of the major goals of SE-DS is to support software managers in selecting suitable SE technologies. Suitability implies the existence of a defined level of evidence about the effectiveness of a specific SE technology in a given context.

In summary, a SE-DS aims at providing managers in the software process with a comprehensive overview of the state-of-the-art and the state-of-the-practice in software engineering in order to facilitate decisions upon new methods and techniques to be introduced. Software engineering roles that might benefit from decision support include the project manager, the product manager, and the quality manager.

2.2 Adaptivity Hypothesis

We set up a pilot study in order to elicit requirements for such a system. We expected that the systems' presentation of the results should be tailored to these different roles. For example, while project managers might be more interested in the impact of a particular method on project results in general, quality managers might have a focus on the potential to detect or reduce error rates. We anticipated a system that might learn from interaction what items to recommend to users in different roles using a case-based reasoning (CBR) approach [1]. Similar user modeling techniques have been shown to be effective in adaptive sales support [4,12] as well as adaptive decision support for IT security tasks [5].

2.3 Data Collection Method

A number of data collection techniques have been suggested to inform the requirements phase when developing a user-adaptive system [2], including interviews, questionnaires, focus groups, systematic observations, task analysis, cognitive and socio-technical models, contextual inquiry, participative evaluation, ethnography.

As we had no existing system or early prototype available that could be analyzed in focus groups or be the basis of a task analysis or contextual inquiry, we decided to assess the preferences and goals of software managers with structured interviews. In order to help interviewees imagine concrete decision support tasks and situations in which a comprehensive SE-DSS might (or might not) be helpful, we offered three scenarios. A scenario consisted of a common part to set the scene of management decision-making (i.e., what kind of information can be obtained, what is the basis for decision support, what is not avail-able), and specific parts linked to one of three particular roles: (1) quality manager, (2) project manager, and (3) product manager.

By introducing these scenarios we aimed at finding differences in the user requirements of the three roles. If that was the case, the development of a comprehensive SE-DSS had to take these differences into consideration by modeling these differences.

The interview questions were developed in collaboration with an expert in cognitive psychology. Question 1 aimed at eliciting reasons for using a comprehensive SE-DSS (motivation). Question 2 aimed at identifying benefits of a comprehensive SE-DSS for improvement management on the organizational level. Question 3 aimed at getting an idea of the amount of user interaction that could be expected. Question 4 aimed at identifying the types of information that users need for comprehensive SE-DS. Question 5 aimed at prioritizing the different types of information needed by the users. Question 6 aimed at getting a better understanding about how query results should be presented to the user. Question 7 aimed at identifying other application areas (not mentioned in the scenarios) of comprehensive SE-DSS.

The questions were not only aimed at eliciting requirements from potential future users of a comprehensive SE-DSS, but also to answer the question if a user modeling approach should be used for implementation. Additionally, the questions were used to substantiate the validity of the scenarios offered to the interviewees.

Obviously, from a user modeling point of view, questions 3, 4 and 5 are the most relevant for creating a user model while we did not expect differences across roles for the other questions.

2.4 Sampling

Participants were selected on the basis of two criteria: In order to be relevant, interviewees had to be sufficiently mature with regards to software management experience. In order to be reliable, a sufficient number of subjects had to be interviewed. Being a research institute that is largely involved in conducting research and transfer projects with software industry, Fraunhofer IESE offered enough experts to conduct a pilot study. In total, seven business area managers, one institute director, and one department head participated in the pilot study. Business area managers are senior consultants who establish and maintain contacts with industrial partners, acquire projects, and help transfer research results into industrial environments. Personal industrial project experience within the group of interviewees ranged from 5 to 17 years.

2.5 Procedure

The interviews were conducted as follows. Interviewees received the common part of the scenario description and two role-specific scenario descriptions a couple of days prior to the interview. When the interview started, first the role-specific scenario was presented to the interviewee. Then, the interviewee was asked to answer the questions from the perspective of the first role. When all questions related to the first role had been answered, the second role-specific scenario was presented to the interviewee,

and the interviewee was asked to take up the second role and think about differences in the requirements for that role. Roughly ¾ of the time were assigned to the first role, ¼ to the second role. Eight of the nine interviews were recorded with an MP3 stick. In one case, a scribe recorded the interview on paper. All interviews lasted between 25 and 35 minutes. The set of questions was not sent to the interviewees in advance. Also, there was no communication between interviewees about the content of the interviews while the study was conducted.

Each interviewee was randomly assigned to two of the three specific roles. Table 1 shows the random assignment of role-specific scenarios to interviewees.

Table 1. Assignments of role-specific scenarios to interviewees (1 = quality manager; 2 = project manager; 3 = product manager)

Interviewee	A	B	C	D	E	F	G	H	J
1st role	1	1	2	1	1	2	1	1	2
2nd role	2	3	3	2	3	3	2	3	3

The procedure we used to aggregate and synthesize the answers given by the interviewees was inspired by the grounded theory approach [11]. We started the transcription with the first interview and the first question. Then we took the next interview and tried to find communalities and differences related to the first answer of the first question. If a similar answer was found, the counter of the first answer to the first question was set from 1 to 2. If no sufficient similarity was encountered, then the new answer from the second interview was added to the list of answers related to the first question. When all interviews were checked for question one, we repeated this procedure for question two, starting with the first interview. When an answer that was given to a question was found to be more related to another question, then this answer was re-assigned to that more relevant question, again following the procedure described above. After having processed all answers related to all questions, we double-checked that the aggregated and synthesized answers still represented sufficiently well the set of answers originally provided by the interviewees.

In addition to counting the occurrence of similar answers, a binary ranking was made: the interviewee explicitly or intuitively expressed high importance (H) of the response to the question, either explicitly ranked it as medium important (M) or did not clearly rank it as highly important. The process of aggregation and ranking resulted in Tables 2 to 6.

2.6 Results

All of the interviewees accepted the pre-defined scenarios as being relevant and practical, none had difficulties with understanding. We interpret this finding to support the construct validity of our measurement instrument (scenario-based structured interviews).

For creating a user model question 4 and 5 are of interest. We found that the participants would actually prefer some kind of intelligent support rather than interacting with a “dull” search engine. Especially in the case of huge result sets it gets attractive to have some support in reducing the number of hits interactively.

However, several people mentioned that a combination of the two strategies might be preferable to give the user the freedom to use the intelligent support or not. Four people mentioned the transparency of the system would be a major issue. Users want to know why they received a specific result set otherwise they would not have enough trust in it. Moreover, the user modeling mechanism should not require them to answer long list of questions, i.e., the interaction should rather be goal oriented. See Table 2 for an overview of the answers and the importance ratings.

Table 2. Interaction preferences (question 3). Frequency of answers with high importance (H) and medium importance (M).

Two alternative interaction strategies. **1. Similar to a search engine but more specialized.** **2. Iterative refinement of the solution area by user model based interaction.** **Which strategy would you prefer, and why?**		**H**	**M**
3.1	A combination of the alternatives is preferable	3	4
3.2	Transparency is important: Why did I get this result set? Access to the full set should be possible	4	
3.3	Not answering lots of questions, but fill in a template with check-boxes	2	
3.4	Especially in case of a huge result set, the second alternative becomes more attractive	3	1
3.5	Guidance for reducing the result set (e.g., use the context to reduce result set)	3	
3.6	Interaction has to be goal/problem oriented	1	

Questions 4 and 5 are closely related. In order to get unbiased but comparable answers at the same time we split the assessment of presentation preferences into two parts. First, we asked openly for the type of information that should be provided by a SE-DSS, i.e., the interviewees brainstormed on information types without guidance in order to not neglect relevant types a priori. Second, each participant had to rate a given list of information types in regard to his or her preferences. We intended to base the design of the user model on this rating.

Question 4 validates our assumptions about the relevant information. Participants mentioned most of the information types that were included in our list anyway but did not request new types. See Table 3 for a summary and relevance rating of the answers.

In question 5 the interviewees were asked to rank the types of information that is available from empirical studies with regard to the value the information delivers them for their decision process.

Tables 4 to 6 show the raw data and the mean of the ratings. There is no clear profile of any of the three roles. The variations of preferences within a role (e.g., comparison of Q1 and Q2) seem to be similar to the variations identified between roles (e.g., Q1 and Pj1). Statistical analysis yielded no significant differences, but is of course limited due to the low number of subjects.

Table 3. Types of information needed (questions 4). Frequency of answers with high importance (H) and medium importance (M) categorized by role.

	Results from empirical studies can be described and aggregated differently. Which information should be provided by the DSS?	**H**	**M**
	Quality manager		
4.1	Which techniques are available (information on a highly aggregated level)?	1	
4.2	How effective/efficient is a certain technique with respect to which quality aspect?	1	
4.3	Description of the process in which a SE technique shall be applied	1	
4.4	Costs for introducing/applying the SE technique	1	
4.5	Experience with the application of the technique	1	
4.6	To get information about the impact a single SE technique has on the whole development process	1	
4.7	Information that allows for conclusions about the validity of empirical results associated with a particular SE technique	1	
4.8	Context information (kind of system, programming language, process step)	1	
	Project manager		
4.9	Description of the project context in which the results are gained	2	
4.10	Costs for introducing/applying the SE technique	1	1
4.2	How effective/efficient is a certain technique with respect to which quality aspect?		1
4.11	Needed level of education of the employees		1
	Product manager		
4.8	Context information (kind of system, programming language, process step)		1
4.12	Description of the technique		1
4.5	Experience with the application of the technique		1
4.13	Who has applied this technique	1	1
4.14	Impact on product quality and development costs		1
4.11	Needed level of education of the employees	1	
4.2	How effective/efficient is a certain technique with respect to which quality aspect?	1	

Table 4. Quality managers' (Q1-Q4) ranking of information types (question 5)

Interviewee & Role	**Q1**	**Q2**	**Q3**	**Q4**		
Kind of Information	**Rating**				**Mean**	**Rank**
Results	11	13	14	14	13	1
Lessons learned	1	14	13	11	12.25	2
Others' Experience	8	14	11	14	11.75	3
Evaluated techniques	12	11	14	8	11.25	4
Validity of results	10	12	13	10	11.25	5
Object of Study	14	0	12	14	10	6
Purpose of evaluation	12	0	12	14	9.5	7
Quality attribute	13	10	14	0	9.25	8
Hypotheses	11	0	13	12	9	9
Subjects	9	2	12	13	9	10
Controlled variables	9	5	12	6	8	11
Dependent variables	9	4	12	7	8	12
Publication	8	3	11	9	7.75	13
Cost (educating the subjects)	0	0	12	14	6.5	14

However, we were not able to demonstrate differences in presentation preferences in regard to the roles.

The seventh question was not intended to elicit new requirements but to confirm the relevance of our scenarios, and to identify new/other application areas for a comprehensive SE-DSS. Since the answers were not used for requirements elicitation,

we omit the related table here. The relevance of the scenarios was confirmed. In addition, the answers confirm findings from question two, but on a more general level. For example, it was mentioned that a comprehensive SE-DSS could be used to educate new employees, or store (and maintain) project experience. Additionally, the available information might be used to focus in future studies on SE technology effectiveness/efficiency, and thus help improve the coordination of empirical research.

Table 5. Project managers' (Pj1-Pj2) ranking of information types (question 5)

Interviewee & Role	**Pj1**	**Pj2**		
Kind of Information	**Rating**		**Mean**	**Rank**
Evaluated techniques	14	14	14	1
Others' Experience	14	14	14	2
Validity of results	13	14	13.5	3
Object of Study	14	12	13	4
Quality attribute	14	12	13	5
Results	14	12	13	6
Controlled variables	13	13	13	7
Lessons learned	13	12	12.5	8
Cost (educating the subjects)	13	12	12.5	9
Subjects	12	13	12.5	10
Purpose	12	12	12	11
Hypotheses	12	12	12	12
Dependent variables	12	12	12	13
Publication	12	12	12	14

Table 6. Product managers' (Pd1-Pd3) ranking of information types (question 5)

Interviewee & Role	**Pd1**	**Pd2**	**Pd3**		
Kind of Information	**Rating**			**Mean**	**Rank**
Object of Study	14	14	14	14	1
Quality attribute	13	14	13	13.33	2
Results	13	13	11	12.33	3
Hypotheses	11	13	12	12	4
Validity of results	13	13	10	12	5
Dependent variables	13	14	9	12	6
Purpose of evaluation	13	14	8	11.66	7
Lessons learned	13	14	7	11.33	8
Others' Experience	13	13	6	10.66	9
Evaluated techniques	13	14	5	10.66	10
Subjects	13	14	4	10.33	11
Controlled variables	12	14	3	9.66	12
Cost (educating the subjects)	13	13	2	9.33	13
Publication	12	12	1	8.33	14

3 Discussion

Surprisingly, we did not find much difference between the three management roles. The answers given were too similar, no matter which specific role was assigned to an interviewee.

3.1 Limitations of the Instrument

At the moment, it is not fully clear whether this indicates that differences between roles are not as large as we originally expected, or whether the answers given by the interviewees were too strongly influenced by the way role-specific scenarios were presented to them. Also, the subjects might not be fully representative for the specified roles due to the nature of their work in research environments, which is probably not as strongly focused on actual (and mostly short-term) decision-making within software projects.

Some of the subjects were skeptical if they would use such a web-based DSS, as they felt unable to estimate the actual power of such a system in supporting them in their job. This partly seems to reflect the common fear that web-based information sources potentially create information overload.

Moreover, users might just have been unable to anticipate what would be adequate functionality under particular circumstances. A scenario can certainly only sketch the potential power and functionality. Design issues are intentionally precluded.

SE-DSS are supposed to provide the basis for important strategic decisions such as the adoption of a new engineering method in a project. In order to do that, users must have trust in the recommendations of the system. The system must respect usability goals such as predictability, comprehensibility and controllability [3]. We argue that study participants might not have been in favor of an adaptive version of the system as it might compromise these usability goals. While user tests and summative evaluations might even show that user-adaptivity can increase performance (i.e., precision and recall) and effectivity, users might be frightened by the possibility of "biased" search results when confronted directly.

3.2 Alternatives

Despite these limitations of the requirements elicitation instrument used in the example above, alternative approaches are not obvious. Scenarios and structured interviews were chosen due to the lack of a prototype or similar system. In the hindsight it might have been worthwhile to invest additional effort in creating a mock-up prototype that illustrates the potential of such a system.

4 Summary

In the presented case study, structured interviews did not elicit sound requirements for adaptivity. Unfortunately, the results remain inconclusive in regard to whether it is worthwhile to model user characteristics in this particular case.

Acknowledgement

Similar data has been presented in a workshop paper by the same authors [6], however, we believe that it might present a valuable contribution to the workshop on User-Centered Design and Evaluation and is thus worth to be reported again.

References

1. Aamodt, A. and Plaza, E.: “Case-Based Reasoning: Foundational Issues, Methodological Variations and System Approaches”. AI Communications Vol. 7, No. 1, 39-59, 1994
2. Gena, C.: “Methods and techniques for the evaluation of user-adaptive systems”, The Knowledge Engineering Review, Vol. 20, No 1, 1-37, March 2005
3. Jameson, A. “User Modeling Meets Usability Goals”, Proceedings of 10th International Conference on User Modeling, UM2005, pp.1-3, 2005, Berlin, Springer.
4. Jedlitschka, A. and Althoff K.-D.: “Using Case-Based Reasoning for User Modeling in an Experience Management System”; Proceedings: LLWA01 Tagungsband der GI-Workshopwoche “Lernen-Lehren-Wissen-Adaptivität“, Forschungsbericht Nr. 763 Universität Dortmund, October 2001
5. Jedlitschka, A.: Adaptive Support for IT Security Tasks. In Ricci, F. & Smyth, B.(Eds.) Proc. of the AH’2002 Workshop on Recommendation and Personalization in eCommerce, Málaga, Spain, May 28th 2002. Universidad de Málaga
6. Jedlitschka, A., Weibelzahl, S. and Ayari, B.: “Is it worthwhile to model users of a tool in support of SE technology selection”. *In: Proceedings of the workshop week Lernen - Wissensentdeckung - Adaptivität (LWA2004)* , October 4 - 6, 2004, pp. 36-41, Humboldt University, Berlin.
7. Katz-Haas, R. “Ten Guidelines for User-Centered Web Design”, Usability Interface, Vol 5, No. 1, July 1998
8. Koch, N.: “Software Engineering for Adaptive Hypermedia Systems: Reference Model, Modeling Techniques and Development Process”. UNI-DRUCK Verlag, 355 pages, 2001. PhD. Thesis, Ludwig-Maximilians-Universität München
9. Ruhe, G.: “Software Engineering Decision Support – A new paradigm for Learning Software Organizations”. In: Proc. Workshop. Learning Software Organizations, Springer, 2003.
10. Ruhe, G.: “Software Engineering Decision Support: Methodology and Applications”. In: Innovations in Decision Support Systems (Ed. by Tonfoni and Jain), International Series on Advanced Intelligence Volume 3, 2003, pp 143-174.
11. Strauss, A. and Corbin, J.: Basics of Qualitative Research. Techniques and Procedures for Developing Grounded Theory. 2nd ed. Thousand Oaks: Sage, 1998.
12. Weibelzahl, S. & Weber, G. (1999). Benutzermodellierung von Kundenwünschen durch Fallbasiertes Schließen. In Proc. 7. GI-Workshop "Adaptivität und Benutzermodellierung in interaktiven Softwaresystemen", LWA-1999, Otto-von-Guericke-Universität Magdeburg, Germany, 29.-30. Sep.1999, 295-300