

# Developer-User Communication in Large-Scale IT Projects

Ulrike Abelein

Institute of Computer Science, University of Heidelberg, Im Neuenheimer Feld 326, 69120  
Heidelberg, Germany  
[abelein@informatik.uni-heidelberg.de](mailto:abelein@informatik.uni-heidelberg.de)

**Keywords:** User-Developer Communication; User Participation, User Involvement, Software Intelligence

**Abstract.** User participation and involvement in software development are considered to be essential for a successful software system. In large-scale IT projects with traditional development methods the end user is mostly involved in the beginning of development (i.e. in the specification phase) and at the end of development (i.e. in the verification and validation phase). But there are also user-relevant important decisions in the phases in between (i.e. design and implementation). Thus, I argue that it is important to study how large-scale IT projects can enhance user-developer communication in order to increase system success. I investigated what evidence exists on effects of user participation and involvement on system success and explored which methods are available in literature and in practice through an interview series. In addition, the thesis will propose a method that supports large-scale IT projects in enhancing user-developer communication. As a first step I developed a descriptive classification containing user-relevant decisions and therefore trigger points to start user-developer communication. Furthermore a tool analysis and extension of one tool will ensure the feasibility of the method in real life large IT projects. Finally the feasibility and effects of the method will be evaluated in a case study.

## 1 Introduction

The complexity and scale of business software (SW) systems, such as enterprise resource planning (ERP) systems, has dramatically increased over the last decades [1]. Large-scale IT projects involve many stakeholders, whose different goals often lead to conflicting opinions and requirements. The resulting software system is supposed to be consistent with the desires of all stakeholders; therefore a need to involve stakeholder and in particular end users exists. In regard to user participation and involvement (UPI), there are three clear advantages of these large scale business system implementations. First, in comparison to new or evolutionary development of systems for a mass market, the prospective users are available within the company as are the developers who work long term on such systems. Second, a clear business

trend towards customized off-the-shell systems (COTS) over individual development of bespoke systems exists. As these standard systems already have best practices functionality built in, this leads to a defocus on actual functionality and a focus on customization. Thus, the technical challenges get less important, but the need to involve end users increases, due to their specific context knowledge. Third, in enterprises implementing these large-scale systems in comparison to mass market software, both end users and IT developer have an interest in achieving system success. The end users' work is dependent on the resulting system and IT personnel often have a hard position within large enterprises as their work is only perceived as support of the main business, thus they have an interest to improve their perception in the business domains.

So far most research on UPI focuses either on early or on late development phases [2], [3]. I believe that the step in software development when user requirements are refined (and thus interpreted) by developers into a technical specification (i.e. system requirements, architecture and models) is a critical one for UPI and specifically for user-developer communication. In this step a lot of implicit decisions are taken, some of which should be communicated to the end users. Thus, within the thesis, I plan to create a method that extends existing requirement engineering, software development and project management practice in order to enhance user-developer communication. Therefore, I identified trigger points (decisions that are made during software development) that initiate communication with the end users, developed a classification of user relevant decisions to define the granularity level on which to communicate with the end users, derived adequate means of communication based on the media richness theory, and will propose a setup enabling large-scale IT projects to enhance user-developer communication. To create the method I first did a synthesis of current research regarding studies of correlations between UPI and their effect on system success. Secondly, an analysis of existing methods for UPI in software development and IT project management helps to identify gaps to be closed with the new method. Third, an interview series has been conducted to validate a classification of user-relevant decisions in the design and implementation phases. Fourth, the method will be detailed based on the interviews and validated in a case study to show the effects on system success.

## **2 Problems and Research Questions**

Most large-scale IT projects are still using traditional project management and SW development methods like the waterfall model [4], [5]. Their advantages are high stability and clear agreements on price, timeline and scope [6]. However, the drawbacks are long periods of waiting for the business side [6]. Within these long development cycles requirements transform, as the translation from user to system requirements leads to a lot of interpretation and misunderstanding accompanied by a low level of user-developer communication. There are two effects: On one hand, end users do not feel integrated in the project. On the other hand, end users do not recognize their requirements in the acceptance phase [7]. Both effects lead to a low

acceptance of the system and a low motivation to participate in large-scale IT projects. In addition, a lot of frustration and inefficiency exists due to communication gaps between the project participants, in particular between the business users and the IT personnel (developers, designers and architects) [8]. Especially the backwards communication of decisions and their rationales after the requirement elicitation does mainly not exist in those projects [9]. A method that enables IT projects to enhance user-developer communication regarding rationale of decisions will help end-users to feel more integrated in the project and thus motivate them to support the project with their knowledge. This will not only lead to a higher system quality, but also a higher acceptance rate and usage of the resulting system.

Thus I want to answer the following questions in the thesis:

- RQ1- Does increased UPI lead to increased system success in large-scale IT project?
- RQ2 - What are the characteristics of existing methods in literature aiming to increase user participation and involvement in software development?
- RQ3 – How can a large scale IT project support user-developer communication (with a focus on the decisions and their rationale that are made in design and implementation phase) in order to increase system success?
- RQ4 – What effects has the method that supports large-scale IT projects in user-developer communication?

### 3 Proposed Solutions

#### 3.1 Solution to RQ1 - Evidence that increased UPI increases system success

I conducted a systematic mapping study (submitted in December 2012 to the Journal for Empirical Software Engineering [10]), in which I identified empirical evidence in surveys and meta studies. I developed an overview of structural equation models that demonstrates that most papers showed positive correlations between aspects of development processes (incl. user participation), human aspects (incl. user involvement) and system success. Within the systematic mapping study, I extracted the researched aspects, correlation and number of participants for validation from 90 studies that were the result of our literature review. In order to analyze the aspects, I developed a classification with the main categories: development process, human aspects, system attributes, organizational factors and system success. The analysis revealed that *user participation and involvement is an important research topic*, as it has been researched in a broad manner by various research areas. The vast majority of the derived correlations showed a positive effect, thus I can conclude that *aspects of the development process and human aspects have a positive effect on system success*. Another indicator for the wide range of this research area is the number of participants that were employed to validate the effects on a subcategory level. The analysis showed that *user participation and their involvement's positive effect on user satisfaction was validated by a more than 4000 participants involved in the surveys*. Users, who feel involved, do use the system more frequently. Lastly, I looked into the

15 studies with negative correlations. Most of them show only a few negative correlations, but do not question the main correlations between aspects of UPI and system success. In addition, I found out that *most studies with negative correlations were published a long time ago.*

### **3.2 Solution to RQ2 - Characteristics of methods increasing UPI**

I looked into 27 methods papers within the systematic mapping study [10] and analyzed their targeted issue, their validation context and their proposed solution. I found out that *all software development activities (planning & project management, SW specification & requirement engineering, SW design & implementation, SW verification & validation, and SW evolution) are influenced by methods, but not many methods focus on the design and implementation activity.* The comparison between aspects researched by the surveys and meta studies and the targeted aspects from the methods reveals that methods for user participation and involvement target similar categories as the surveys and meta studies. But they do have a higher focus on the user-developer communication and the user's motivation. In addition, they target mostly the success factor system quality, which differs from the survey papers that mostly research user satisfaction. The analysis of the validation context revealed *that most methods were validated in a public environment.* The structured overview of practice with method examples shows *that practices derived from the solutions are distributed over all software activities.* In addition, I identified a focus on communication structures in the methods.

### **3.3 Solution to RQ3 – Support of user-developer communication**

I will propose a method to support large-scale IT projects in enhancing user-developer communication with the four components: setup of communication structure based on stakeholder analysis, train developers on capturing decisions/changes, setup traceability of decisions, and define means of communication based on media richness theory. An interview series with twelve experts in large scale IT projects has been done. The results will be used to identify whether there is communication between end users and developers in large scale IT projects and if yes in what setup and phases it takes place. Furthermore, issues and consequences that are caused by communication gaps will be identified. I collected 81 examples of trigger points and thereof developed a descriptive classification for user-relevant decisions. The idea for a method has also been validated and will be improved and detailed..

### **3.4 Solution to RQ4 – Effects of method for user-developer communication**

To assess the feasibility and effects of our method I will use the results of the interview series as well as the case study. I want to rate the usefulness of our method structure and hope to be able to measure improvements in user satisfaction when

testing the feasibility of the method within the case study. In addition, I will look into effects, such as is there an increase of direct user-developer communication and general communication interactions in the design and implementation phase after implementing the method?

### **3.5 Expected Benefits**

Overall, the meta analysis showed a positive effect of UPI and in particular of user-developer communication on system success. Thus, a theoretical base of our method is available. The results give insights specifically for the community of human aspects in software engineering into the existing research on UPI. The overview of existing methods is useful for other researchers as they can see what method covers which targeted issues in which context and what the proposed solution is. In addition, it aids to understand the landscape of software development and IT project management methods in regard to UPI and indicates which parts of existing methods could be reused and combined in the new method. The descriptive classification of user-relevant decisions supports the method by helping developers and end users to understand important decisions and their implications. Through the interview series, I validated the classification and enhanced it with examples. The examples of decisions will help to explain, users, developers and researcher to understand when to start communication with the end users. The method will help to close communication gaps especially for the area of large-scale IT projects in a business context. There is a large bandwidth of existing methods, but a low usage rate within practice. Thus, it seems to be very hard to find the right balance between the developers' and end users' division of work and close alignment between these parties with a high level of communication. The method will describe in what situations it is useful to start communication with the end user (trigger points), how to structure that communication (when to inform whom on what granularity level) and how to represent these decisions and the rationale to help the end user to understand them. The validation of the method within a case study will also point to open issue and refinement needs within the method and rate the benefits.

## **4 Research Methods**

The literature review is conducted as a systematic mapping study [11]. A search string has been used in twelve different sources from the domains (IT, Business and Communication). Overall 3136 hits have been identified, the initial selection based on publication title and abstract lead to a 232 downloaded publications. This selection has then been reviewed with clear exclusion criteria. The classification for end-user-relevant decisions is developed based on analysis of existing methods and then targeted for the context of large scale IT projects (RQ 3). The interview series was conducted with semi-structured interviews of twelve experts in large-scale IT projects. The evaluation of the method depends on the company for the case study. At

least one very large case study comprising a questionnaire of current situation and usage of user-developer communication, application of our method in (one or more) IT projects and analysis of implications on project success from the applied method will be done. Ideally this case study will be conducted by accompanying a real life IT implementation over a longer time period.

## **5 Related work**

So far the topic of user participation in IT projects has mainly been researched in the information system field. This research mostly focused on the work-place context and looked for dependencies of UPI and system success. A broader approach has been taken by the research area of human-computer-interface. This area focuses on the design of interactive systems and their usability mostly known under the words of 'user-centered' or 'user-centered design' [2]. User-centered design utilizes methods such as task analysis, prototyping and usability evaluations [12]. Other forms are participatory design - focusing on democratic participation through workshops -, ethnography – emphasizing social aspects through observation-, and contextual design – looking into the context of work through contextual inquiry prototyping [13]. Within software engineering the topic has been of much interest, as neither user participation nor user involvement is mentioned in the SWEBOK [2]. Despite this amount of existing research there are still gaps within the different methods and it is still an open question how user involvement should be integrated into SW development [14], [15]. Other methods such as participatory design based on the Scandinavian school, user-centered design defined in the ISO standard or joint-application-design [16] fail to point out how exactly (i.e. in which phases, which content, etc) the user involvement should take place [2]. So far most research focus on UPI either in the early development phases, e.g. requirement elicitation, or at the end of the development project within user acceptance tests [2], [3]. An interesting study has been done by Bjarnason et al. (2011) [8]. They study communication gaps in terms of their (root) causes and effects (e.g. customer expectations that are not met, low motivation to contribute to the requirements work, software unit control of the implementation without alignment with the requirement team, unclear requirement coverage, quality issues and wasted effort from rework). Given those effects I believe that the step in development process when the user requirements are translated by the developer into the more technical specification of the system is a critical one. Even though most agile approaches implicitly use that sort of communication as they claim very close cooperation (mostly even physically together in one team room), the focus is more on a successful way to quickly develop working software. Besides that, these methods are hard to implement in large-scale IT projects. It is still an open question how the current high-ceremony methods can be extended by agile methods [5], as in most of the long term implementation projects the end users from business side cannot be a full time team member (as they need to perform their daily work).

## 6 Progress

I started the work on the PhD in September 2011. In the autumn of 2011, I have conducted the literature review and got familiar with the topic of user-developer communication. In 2012, I developed the first structure of the method based on TORE and the Media Richness Theory. Furthermore, I used the results of the systematic mapping study in order to answer RQ1 and RQ2. In addition, I conducted an interview series with twelve experts on large-scale IT projects. In 2013, I will analyze the results of the interview series and will detail the method. I will also include a tool analysis. Finally I want to conduct the case study for validation. I expect to finish the thesis by December 2013.

## 7 References

- [1]S. Kanungo and S. Bagchi, "Understanding User Participation and Involvement in ERP Use," *Journal of Management Research*, vol. 1, no. 1, pp. 47–64, 2000.
- [2]J. Iivari, H. Isomäki, and S. Pekkola, "The user - the great unknown of systems development: reasons, forms, challenges, experiences and intellectual contributions of user involvement," *Information Systems Journal*, vol. 20, no. 2, pp. 109–117, Mar. 2010.
- [3]B. Ives and M. Olson, "User involvement and MIS success: a review of research," *Management science*, vol. 30, no. 5, pp. 586–603, 1984.
- [4]R. D. Austin and R. L. Nolan, *How to manage ERP initiatives*. Boston: Division of Research, Harvard Business School, 1998.
- [5]G. B. Alleman, "Agile project management methods for ERP : how to apply agile processes to complex COTS projects and live to tell about it," in *Extreme Programming and Agile Methods: XP/Agile Universe*, D. Wells and L. Williams, Eds. Springer Verlag, 2002, pp. 70–88.
- [6]M. Fowler and J. Highsmith, "The agile manifesto," *Software Development*, vol. 9, no. August, pp. 28–35, 2001.
- [7]W. J. Doll and G. Torkzadeh, "A discrepancy model of end-user computing involvement," *Management Science*, vol. 35, no. 10, pp. 1151–1171, 1989.
- [8]E. Bjarnason, K. Wnuk, and B. B. Regnell, "Requirements are slipping through the gaps — A case study on causes & effects of communication gaps in large-scale software development," in *2011 IEEE 19th International Requirements Engineering Conference*, 2011, pp. 37–46.
- [9]A. Al-Rawas, S. Easterbrook, U. S. N. Aeronautics, and S. Administration, "Communication problems in requirements engineering: a field study," *COGNITIVE SCIENCE RESEARCH PAPER-UNIVERSITY OF SUSSEX CSRP*, no. February, pp. 1–2, 1996.
- [10]U. Abelein and B. Paech, "Understanding the Influence of User Participation and Involvement on System Success - a Systematic Mapping Study," *Journal of Empirical Software Engineering*, p. Submitted in Dec 2012, 2012.
- [11]B. Kitchenham and S. Charters, "Guidelines for performing systematic literature reviews in software engineering," *Engineering*, vol. 2, no. EBSE 2007–001, 2007.
- [12]M. A. Harris and H. R. Weistroffer, "A New Look at the Relationship between User Involvement in Systems Development and System Success Development and System Success," *Communications of the Association for Information Systems*, vol. 24, no. 1, pp. 739–756, 2009.

- [13]S. Kujala, "User involvement : a review of the benefits and challenges," *Behaviour & Information Technology*, 2003.
- [14]T. Lynch and S. Gregor, "User participation in decision support systems development: Influencing system outcomes," *Eur J Inf Syst*, vol. 13, no. 4, pp. 286–301, Nov. 2004.
- [15]S. R. Humayoun, Y. Dubinsky, and T. Catarci, "A Three-Fold Integration Framework to Incorporate User – Centered Design into Agile Software Development," *Work*, vol. 6776, pp. 55–64, 2011.
- [16]J. Wood and D. Silver, *Joint application development*. New York: John Wiley & Sons, 1995.