Empir Software Eng
DOI 10.1007/s10664-013-9278-4

Understanding the Influence of User Participation and Involvement on System Success – a Systematic Mapping Study

Ulrike Abelein · Barbara Paech

Abstract User participation and involvement in software development are considered to be essential for a successful software system. Three research areas, *human aspects of software engineering*, *requirements engineering,* and *information systems,* study these topics from various perspectives. We think it is important to analyze user participation and involvement in software engineering comprehensively to encourage further research in this area. We investigate the evidence on effects of user participation and involvement on system success and we explore which methods are available in literature. A systematic mapping study was conducted. The systematic search yielded 3,698 hits, from which we identified 289 unique papers. These papers were reviewed by the first author based on inclusion and exclusion criteria. The second author validated the selection of papers by reviewing the reasons for exclusion and inclusion and the corresponding papers on a sample base. 58 of the 289 papers were selected (22 statistical survey and meta-study papers and 36 methods papers). Based on the empirical evidence of the surveys and meta-studies, we developed a meta-analysis of structural equation models. This overview demonstrates that most papers showed positive correlations between aspects of development processes (including user participation) and human aspects (including user involvement) and system success. The analysis of the proposed solutions from the method papers revealed a wide variety of user participation and involvement practices for most activities within software development.

Keywords User participation · User involvement · Software development · Systematic mapping study · Literature review · Meta analysis

Communicated by: Daniela Damian

U. Abelein (✉) · B. Paech
Institute of Computer Science, University of Heidelberg,
Im Neuenheimer Feld 326, 69120 Heidelberg, Germany
e-mail: abelein@informatik.uni-heidelberg.de

B. Paech
e-mail: paech@informatik.uni-heidelberg.de

1 Introduction

User participation and involvement (abbreviated with UPI in case we want to refer to both terms together) in software development are considered to be very essential for system success (Harris and Weistroffer 2009). Users are an important source of information as they are familiar with the work and the context which the software system should support (Hendry 2008). Therefore, involving users is critical to develop good solutions (Hope and Amdahl 2011). The benefits of UPI are very widespread. A summary from (Harris and Weistroffer 2009) names advantages, such as improved quality due to more precise requirements and the prevention of unneeded, expensive features. Furthermore, users who feel involved in a software system will have a positive attitude and perceive it as more useful, thus they are more satisfied with the system (McGill and Klobas 2008). As the level of user acceptance as well as the understanding of the system rises, the system will be used more effectively, and the increased participation in decision-making leads to a more democratic organizational culture. These benefits can all increase the success of a software system. System success is controversial and difficult to measure (Harris and Weistroffer 2009). We want to include all aspects of system success that have been investigated by the identified papers and therefore categorize them in various aspects for system success, such as user satisfaction, ease of use, system use, system quality, data quality, and project in time and budget. We intend to emphasize the broadness of success in this paper, therefore we define system success as the "assessment, whether the information technology project and the resulting system has achieved its objectives".

In general, there are three different research areas that study aspects of UPI: *information systems, human aspects* of software engineering and *requirements engineering*.

So far, the topic of UPI in software development has mainly been researched in the *information system* field. (Iivari 2004) explains this by the distinctive activities in information systems development, where the alignment of information technology (IT) artifacts, organizational and social context is crucial. This research mostly studies empirical dependencies between UPI and system success. Most of the analyzed studies use structural equation models (SEM) to present the identified aspects and correlations.

Within software engineering, this topic has not received much attention. For example, this is evident by the fact that neither user participation nor user involvement is mentioned in the SWEBOK (Iivari et al. 2010). However, the relatively new field of *human aspects* within software engineering (for detailed definition see Hazzan and Tomayko 2004) aims to understand the needs of people involved in software development processes and to support their activities. It also considers the fact that a lot of defects in software are the results of human mistakes (Hazzan and Tomayko 2004). Furthermore, *requirements engineering*, as a subfield of software engineering, concentrates on eliciting requirements from end users and other stakeholders in order to understand the functionality a software system should fulfill (Sommerville 2007).

Even though a lot of studies state the positive effects of UPI on system success, some indicate contradicting results (e.g., Cavaye 1995; Olson and Ives 1981). In addition, there are still a lot of problems with UPI in software development projects. Especially large-scale projects, using traditional software development methods (Alleman 2002), use UPI on a very limited basis. Therefore, we believe that new methods for UPI in software development need to be developed.

In order to encourage further research on UPI methods, two areas should be explored.

First, the clear link that increased UPI leads to increased system success needs to be established. We believe a comprehensive meta-analysis with data from all research areas will increase transparency and help researchers, who want to develop new methods, to ensure the positive effects of their methods. This will also help researchers who want to study aspects of UPI and system success or corresponding context factors further.

Second, it is important to summarize and analyze existing methods of UPI in order to understand the existing method landscape and its limitations (e.g., what software development activities are not covered by existing UPI methods). Such an overview will help other researchers to identify gaps for new methods. In addition, a summary of UPI methods can support practitioners, who want to use UPI in software development, to identify existing methods for their context.

As the effects of UPI on system success, as well as methods of UPI have been studied for a long time, some meta-studies on UPI do exist, (McKeen et al. 1994; Cavaye 1995; McGill and Klobas 2008; Harris and Weistroffer 2009; Kujala 2003; Ives and Olson 1984). Furthermore, a very recent work (Bano and Zowghi 2013) also systematically reviewed the relationship between user involvement and system success. This work has been conducted in parallel to our study and therefore is not included in our meta-analysis. We include a discussion of how this study extends and differs from our work in the strengths and weaknesses section of this paper. However, none of them provides a comprehensive overview combining qualitative and quantitative data and considers both the information systems and the software engineering field.

Therefore, we decided to undertake a systematic mapping study (Kitchenham and Charters 2007) in the field of UPI within three different domains. The strength of this systematic mapping study is the wide scope in which we consider the influence of UPI in software development. We analyze statistical surveys and meta-studies and synthesized their correlation data in a meta-analysis. We complement our study with a description of various methods, which we analyzed and identified practices from. In total, we analyze the results of 58 scientific papers in this systematic mapping study. In addition, we use the secondary data of the six meta-studies on the effects of UPI, which comprise the data of 64 additional studies.

In Section 2, we define important terms and briefly explain approaches to UPI. In Section 3, we sketch the reasoning behind the study and motivate our research questions. Afterwards, in Section 4, we describe our research method and process, followed by Section 5 on the included and excluded studies. In Section 6, we present our results of the surveys and meta-studies, as well as the methods papers. In Section 7, we discuss the results with their strengths and weaknesses and we conclude with an outlook on further research in Section 8.

2 Background of User Participation and Involvement

In this section, we define important terms and explain different approaches for UPI. We start with a definition of user participation and involvement, followed by known approaches for UPI from literature. We present how user participation is covered within various software development methods. At the end, we explain commonly used aspects of system success.

User Participation and Involvement (UPI) The terms "user participation" and "user involvement" are often used interchangeably, but there are also publications that distinguish between them. In our study, we use the two separate definitions of (Barki and Hartwick 1994). Thus, we define *user involvement* as a "psychological state of the individual, defined as the importance and personal relevance of a system to a user" and *user participation* as "behaviors and activities users perform in the system development process". User participation takes place when the end user takes an active part in the development or design process together with the designer (Hope and Amdahl 2011). Even though we defined user participation and user involvement separately, both aspects influence system success (McGill and Klobas 2008). In addition to the distinction of user participation and user involvement, (Hartwick and Barki 1994) have identified several context factors for UPI, such as the characteristics of the system and organizational factors. Furthermore, various aspects of user involvement, such as a user's motivation or a user's attitude towards the system, have been identified.

Approaches to User Involvement and Participation There are many approaches to UPI in literature and practice. We will introduce the main approaches (participatory design, user-centered design, ethnography, and contextual design), as they are necessary for the purpose of this paper (Kujala 2003). The major difference between those approaches is how active the users are and whether they actually participate in decision-making (Kujala 2008). *Participatory design*, originated in Scandinavia, emphasizes democracy and skill enhancement, but efficiency, expertise, quality, commitment and buy-in also have been named as motives (Kujala 2003). It is therefore essential, that users are part of the decision-making process, e.g., in workshops or through prototype evaluation. *User-centered design* comes from the research area of human-computer-interface (HCI). It puts the user, instead of technical needs, into the center of design. Therefore, the designers focus on the users' context (Kujala 2003). In this approach, users are normally not involved in decisions concerning the design; here, other methods such as task analysis are used. *Ethnography* targets the social aspects of human cooperation and uses observations or video-analysis, thus users are involved rather passively. *Contextual design* focuses purely on the context of use for the system and methods, such as the contextual inquiry, which combines observing and interviewing (Kujala 2003).

User Participation in Software Development Methods In addition to specific approaches to UPI, several software development methods exist, that have various options to include users. We briefly introduce the three methods traditional development approaches, rapid application development and agile methods, as they have different approaches to include the user and are the most commonly used methods in practice. *Traditional development approaches,* such as the waterfall model, normally require the determination of a complete, consistent, and accurate list of system requirements, before design and implementation start (Berger 2011). Even though documentation is thorough in this approach, it is often not given to the users. Therefore, users are typically involved only in the requirements definition and the validation process of the system. *Rapid application development* consists of the phases requirements planning, user design, construction, and cutover. Its main advantage is development speed. The short cycles ensure a close match of the system with the business needs, and UPI is normally done throughout the design and development by the evaluation of prototypes (Dean et al. 1998). Nevertheless, this approach has its disadvantages, such as the lack of code reuse or missing consistency of programs. This prevents this methodology from being used in large-scale projects. *Agile or lightweight development* approaches have evolutionary and incremental life cycles and use iterative development and intensive involvement of stakeholders. These methods want to embrace unstable business needs and use flexible development and short implementation cycles to mitigate risks (Berger 2011). Examples are the Dynamic System Development Methodology, SCRUM, or Extreme Programming (XP) (Hope and Amdahl 2011). Some of these approaches demand end users on site and continuous feedback to the end user is required throughout.

It is important to mention that these development methods are not clearly orthogonal to the methods of UPI. For example, it is possible to use participatory design within traditional development approaches (Wagner and Piccoli 2007), but it can also be combined with agile methods.

System Success Aspects in Software Development System success is a controversial issue and difficult to measure (Harris and Weistroffer 2009). Most researchers define system success as the considered aspect that is measured in their study, e.g., user satisfaction, system use or system quality (Hwang and Thorn 1999). It can be measured from a quality perspective, e.g., in *System Quality* based on the ISO 9126 definition (ISO 9126–1, 2011) or *Data Quality*. Another more narrow perspective is the project view, where finishing the software *Project in* a preset *Time and Budget* is important (Chang et al.2010). A commonly used model is the technology acceptance model (TAM), in which the focus is on *System Use* (Davis et al. 1989), which is mainly influenced by the user's perception of *Ease of Use* and usefulness. The perceived usefulness, i.e., the user's degree of favorableness with respect to the system (Wixom and Todd 2005), is often referred to with the term *User Satisfaction.*

3 Research Questions

In order to encourage further research on aspects of UPI and methods of UPI in software development, we want to investigate two areas. First, it is important to strengthen the confidence in existing evidence that UPI has a positive effect on system success. In particular, we wanted to find out which studies with statistical evidence have been published so far and what aspects of UPI and system success as well as further evidence on context factors have been looked into. This will help researchers to ensure that new methods have a positive effect. Furthermore, it helps to identify which aspects are important for new methods. As empirical studies on these effects have mainly been studied in the area of IS this can be most valuable for researchers in the software engineering domain. Second, we want to synthesize existing methods of UPI in software development. This can help to identify gaps for new methods and help practitioners to find a suitable method. Therefore, we want to answer the following research questions:

RQ1: Does increased UPI lead to increased system success?

In order to give a comprehensive overview of the existing evidence and help follow researchers to understand the underlying details in a systematic matter, we broke it down into the following subquestions:

- RQ 1.1 Which aspects of UPI and system success have been looked into by existing studies?
- RQ 1.2 Which correlations between these aspects have been studied?
- RQ 1.3 What are the characteristics of those correlations (percentage of studies reporting positive or negative correlations, variation, amount of participants involved)?
- RQ 1.4 Which further evidence on context factors and their correlations to the aspects of UPI and system success is reported?

RQ2: What are the characteristics of methods which aim to increase UPI in software development?

- RQ 2.1. Which methods do exist to increase UPI in software development projects?
- RQ 2.2. Which activities in software development are affected by these methods?
- RQ 2.3. Which aspects of UPI and system success, as well as context factors, do these methods influence and target?
- RQ 2.4. What are the validation context and the proposed solutions that these methods report on?

4 Review Method

In this section, we explain the different steps of our review method and refer to our exclusion criteria. Detailed definitions of the inclusion and exclusion criteria can be found in Section 5. In order to answer the research questions (see Section 3), we started a systematic mapping study. Following the recommendation of (Kitchenham and Charters 2007), a systematic mapping study needs to have the following characteristics:

C1 a defined search strategy
C2 a defined search string, based on a list of synonyms combined by ANDs and ORs
C3 a broad collection of search sources
C4 a strict documentation of the search
C5 quantitative and qualitative papers should be analyzed separately
C6 explicit inclusion and exclusion criteria
C7 paper selection should be checked by two researchers.

In our systematic mapping study, we followed all of these characteristics, which we indicate with Cx in the following steps of our review method and in Section 5.

We analyzed the surveys and the papers on methods in two separate branches A and B. Branch A represents the statistical survey papers that research correlations of UPI on system success (see RQ 1) and branch B represents the papers on methods that suggest forms of UPI in software development(see RQ 2). We reviewed according to a structured defined search strategy (C1) with an initial phase of three steps (1–generation of search string, 2–identification of research, 3–first exclusion round) and a refinement phase with two steps (4–second round of exclusion, 5–consolidation of results) for each branch. An overview of the research method including the data of the results is shown in Fig. 1.

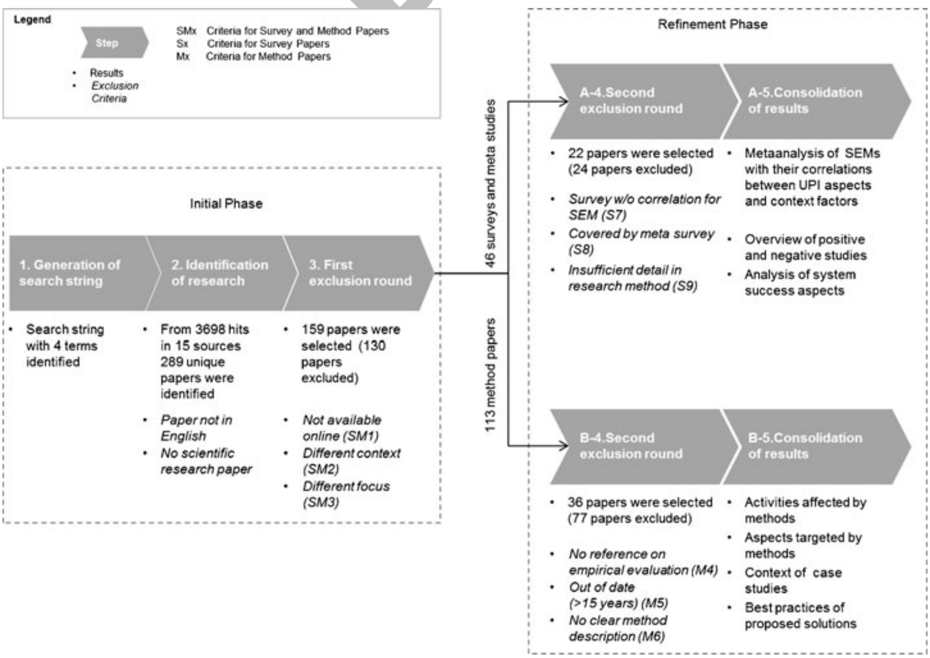

Fig. 1 Overview of the research method

STEP 1 Generation of the Search String

Papers relevant for our research questions needed to cover the key terms for UPI in software projects: end user, participation and involvement, and information technology or software engineering. To identify commonly used synonyms of these terms, we reviewed six sources (Al-Rawas and Easterbrook 1996; Bjarnason et al. 2011; Curtis et al. 1988; Maalej et al. 2009; Maalej and Pagano 2011; Sutcliffe et al. 1999). Based on this review, we developed a search string consisting of four terms (C2). The first term in the search string is a collection of various synonyms for end users. The second term ensures synonyms of participation and involvement. The third term "Participatory Design" is defined by the Scandinavian School (e.g., Kujala et al. 2005) and another often used synonym for UPI. All of these terms had to appear in the title of the paper, as otherwise the probability of identifying relevant research would not have been very high. Searching the abstracts in addition to the titles would have led to too many hits given the amount of sources that we considered. The fourth term was used to ensure that the research was in the context of information technology or software engineering. Therefore, this term was not restricted to the title, but could appear in the full text of the paper. The final search string is illustrated in Fig. 2.

STEP 2 Identification of Research

We want to create a comprehensive picture of the area of UPI; therefore, we used different kinds of sources (electronic sources, general databases, and reference search) in three different domains: information technology, business, and communication (C3). We consciously searched different domains in order to find studies from different perspectives and research areas for our review. The socio-technical perspective was covered by the information technology domain, the management science perspective by the business domain and the social science perspective by the communication domain.

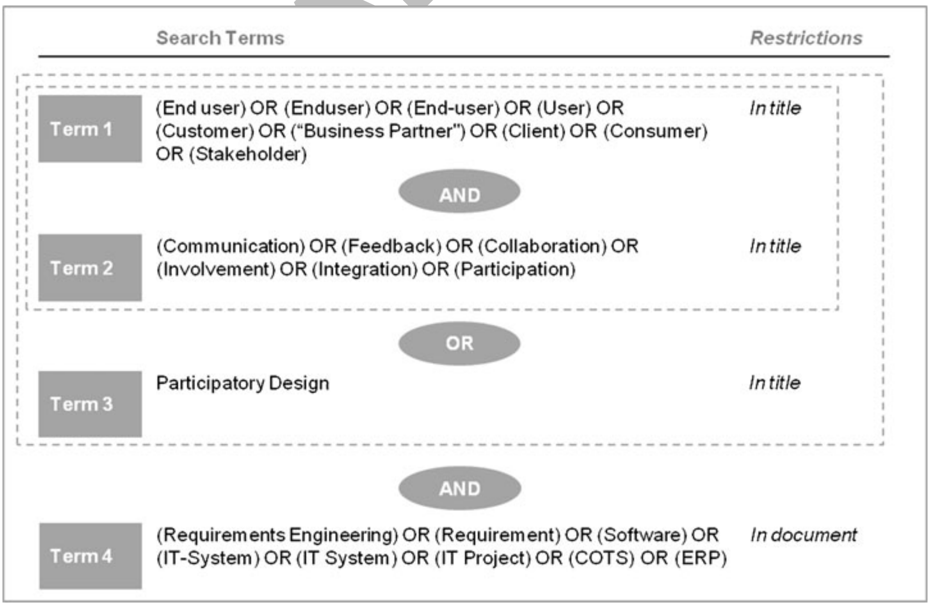

Fig. 2 Derived search string

To identify research, we searched with our search string defined in step 1 in domain-specific publication sources. Respectively, *IEEE, ACM, SpringerLink* for the information technology domain; *MIS quarterly, Harvard Business Review* for the business domain; and the *European Journal of Communication, Research Journal of Communication* for the communication domain. Furthermore, we wanted to ensure that studies from less dominant sources are included in our review, therefore we included four databases covering all domains after the search in domain-specific sources for the search: *Web of Science, Science Direct, Business Source Premier, and Scopus.* In addition, we did a reference ("cited-of") search of (Al-Rawas and Easterbrook 1996) with Google Scholar, as this is a highly cited paper on our initial research topic 'communication of requirements in software development,' and we looked into the sources specializing in UPI: *Information Technology & People Journal, Participatory Design Conference Proceedings* and the *Scandinavian Journal of Information System*. As these sources are very specific to the topic of user participation or involvement, the search string defined in step 1would not necessarily identify relevant papers. Therefore, we adapted the search string for these sources to more general terms and searched for 'Software Engineering' or 'System Engineering' or 'Software Development' or 'System Development' in title, keywords, or abstract. In total, the search retrieved 3698 hits (3393 with the search string, 97 in the reference search, 208 in sources specializing in UPI) in these 15 sources. Following the recommendation of (Kitchenham and Charters 2007), we did an initial selection based on publication title and abstract which lead to 431 results and 289 (250 with the search string, 11 in the reference search, 28 in sources specializing in UPI) unique publications. An indication for the validity and the wide coverage of the source selection is given by the increased number of duplicates in the later searched databases, e.g., the search in the Business Source Premier resulted in 433 papers that had already been identified. We strictly documented the search results, for an overview of the hits, the relevant and unique papers per sources see Fig. 3 (C4).

STEP 3 First Exclusion Round

As 289 publications are too many for a thorough analysis, we conducted a first exclusion round based on the abstract, introduction, and conclusion sections of the papers. We consciously included the conclusion section, as the quality of information technology and software engineering abstracts is not very high. In the first exclusion round, papers were excluded if they were out of the context of software engineering or development. For example, some papers considered UPI in civil engineering or in the development of health care products. Also papers with a different focus than UPI within software engineering were excluded. Lastly, some papers could not be accessed online or in an offline library. 130 papers were excluded based on those criteria (for detailed description see Section 3). The remaining 159 papers were separated in the two branches A (46 papers) for statistical surveys and B (113 papers) for methods of UPI.

STEP 4 Second Exclusion Round

In branch A, 24 papers were removed from our selection in the second exclusion round. These papers did not report clear correlations usable for the meta-analysis of structural equation models (SEM), they were covered by one of the meta-studies, or the research method description did not give sufficient details and therefore were a threat to validity.

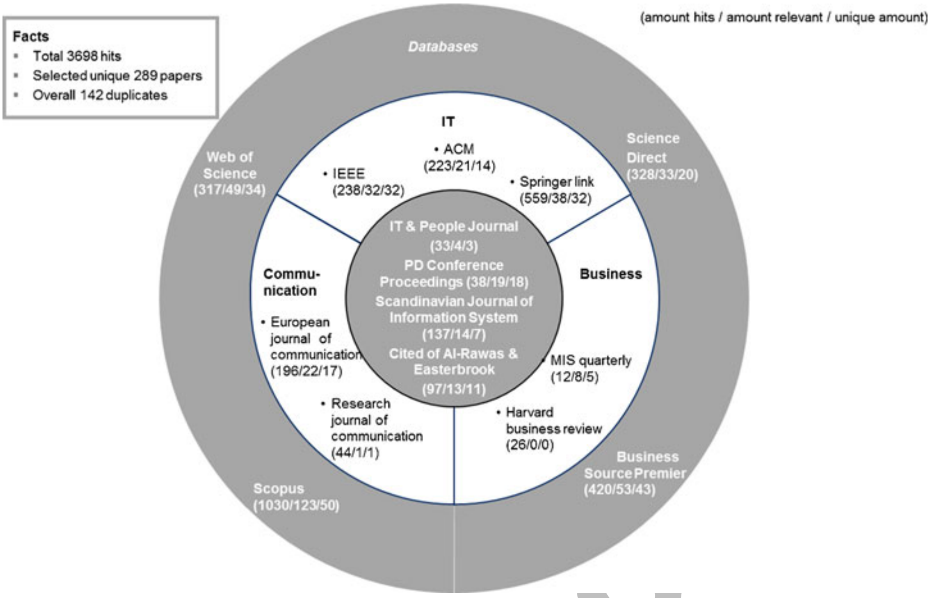

Fig. 3 Sources and hits of the identification of research step

In branch B, 77 papers were excluded, if they did not provide an empirical evaluation of their methods, if the research was out of date (for methods papers, we argue that software and development processes have significantly changed within the last 15 years), or if there was no clear description of a method.

STEP 5 Consolidation of Results

We synthesized the two branches (surveys and methods papers) separately (C5). We compared the results in the categories and subcategories of aspects and context factors (see Section 6.2.). The 22 papers that statistically examine effects of UPI on system success (branch A) are summarized in a meta-analysis of SEMs. Furthermore, an overview of positive and negative studies is presented. The 36 papers of methods (branch B) were analyzed regarding the affected activities in software development, the targeted aspects of UPI, and their validation context. In addition, an overview of the practices of UPI was developed according to the solutions suggested therein. The results of the various analyses can be found in Section 6.

5 Included and Excluded Studies

We defined explicit inclusion and exclusion criteria in order to derive our set of studies (C6). We want to give a comprehensive picture of the influences of UPI in the area of software development. Therefore, we included every paper that either statistically investigates correlations of UPI to system success or that describes a validated method of UPI in software development. In addition, we only included scientific research papers published in English. The papers had to be found in the identification of the research step and not to be excluded by any of the exclusion criteria described below. The list of included and excluded papers was discussed and reviewed on a sample base (about 10 % of the results) for validity by the second author of the paper (C7).

As described in the previous section, we split up the studies into two different categories (survey papers and methods papers). In order to indicate which criterion was used for which category, we abbreviated exclusion criteria that are valid for both categories to SMx and used them in the first round of exclusion. Criteria that are only valid for one category are abbreviated to Sx for survey papers and Mx for methods papers. In total, nine exclusion criteria were defined:

SM1–Not available in library or online

From the 289 papers that have been rated as relevant, based on title and abstract, four could not be retrieved from any (online or offline) library to which we had access. Two of them were published in very small conferences and were therefore neglected.

SM2–Out of context

Papers that describe research on UPI, but within a different context than software engineering, e.g., a different industry such as health care products or a different business function such as marketing were excluded. We excluded 41 papers based on that criterion.

SM3–Different focus

Papers that describe research on users in the context of software engineering and development, but with the focus on a different area than UPI, e.g., improvements of usability of user interfaces, integration of business processes or project portfolio selection, were excluded. We excluded 85 papers based on this criterion.

M4–No empirical validation

Methods papers which did not evaluate their work in a case study and/or a survey were excluded. We excluded 29 papers based on this criterion.

M5–Out of date (published more than 15 years ago)

Only for methods papers, we argue that software and development processes have significantly changed within the last 15 years. Therefore, papers that were published before 1997 were excluded. This is also suggested as a practice exclusion criteria by (Kitchenham and Charters 2007). We excluded 16 papers based on this criterion.

M6–No clear description of methods

Method papers which did not describe a clear method of UPI were excluded, e.g., if the papers only describe high level lessons learned. In addition, method papers which did not describe the method in a detailed manner were excluded. We excluded 32 papers based on this criterion.

S7–Survey without correlations for meta-analysis of SEMs

Survey papers which did not describe correlations of UPI with system success could not be used for the meta-analysis and were therefore excluded. We excluded twelve papers based on this criterion.

S8–Covered by meta-study

If a survey paper was covered by one of the six meta-studies, it was excluded of the selected papers. Nevertheless, we included all the available correlation data of all studies covered by the meta-studies for the meta-analysis of SEMs. This exclusion criterion is important to prevent multiple publication of the same data, as this would bias the results (Kitchenham and Charters 2007). We excluded eleven papers based on this criterion.

S9–Insufficient details in research method

One paper was excluded as it does not give sufficient details about the research method used. In this case, the paper did not clearly describe the people interviewed. This is in line with (Jorgensen et al. 2005).

An overview of the amount of included and excluded papers can be found in Fig. 4.

6 Results

In this paper, we draw a comprehensive picture of influences of UPI in software engineering and development based on our research questions defined in Section 3. The interpretation of the results for each subquestion is presented in Section 7.1. This section is structured in two subsections, one focusing on the statistical surveys and meta-studies and the other on the identified methods papers.

6.1 Results of the Statistical Surveys and Meta-Studies

As suggested by (Kitchenham and Charters 2007) for quantitative studies, we combined the data from the surveys using meta-analytic techniques. This increases the likelihood of detecting real effects that individual, smaller studies are unable to detect. In particular, we wanted to figure out: does *increased UPI lead to increased system success (RQ1)?*

We therefore extracted the SEM and/or the correlations from the 18 surveys as well as the six meta-studies. From the meta-studies we extracted the data from another 64 surveys described in the studies. In total, data from 86 unique studies was used for the overview structural equation model. To answer our research questions 1.1 to 1.4, we extracted the researched aspects of UPI and system success and other context factors, as well as the statistically significant correlations between two aspects or factors. In addition, we extracted the number of participants of each survey wherever possible. In the rare case that we did not find the referred paper of a meta-study, we used 1 or −1 as a replacement for the correlation value, but did not consider that value for the variation of correlations. If we could not find the number of participants, we ignored the study for the analysis. The context factors that influence UPI or system success (RQ 1.4) have been analyzed as a side product of our main research question (RQ 1). We therefore integrated these results in the following subsection on

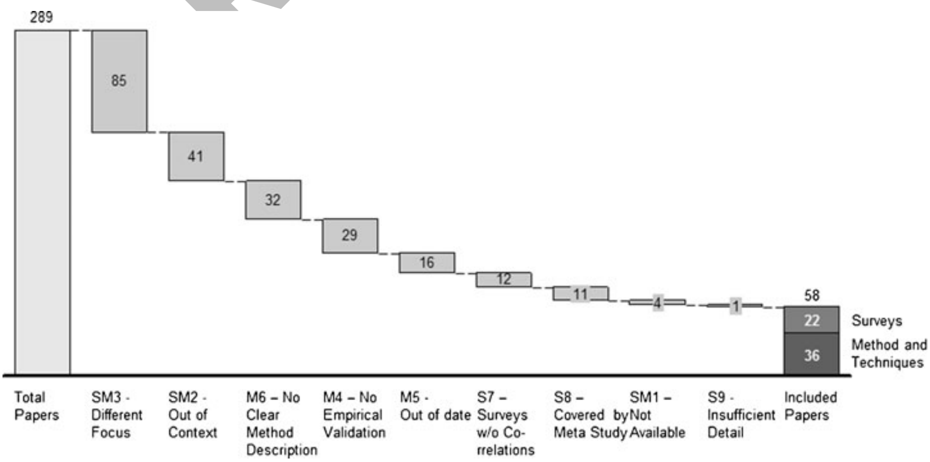

Fig. 4 Overview of excluded and included papers based on criteria

RQ 1.1 to 1.3. In general, it must be mentioned that from a statistical perspective, correlations do not have a direction as they are a dimensionless measure between two values. Nevertheless, it is common practice to set up hypotheses with directed links and then interpret the results in the direction most likely. For our analysis, we used the directed links suggested by the analyzed paper.

6.1.1 Aspects of UPI, System Success and Context Factors (RQ 1.1 and 1.4)

In order to figure out which aspects of UPI and system success, as well as related context factors have been looked into by existing studies, we developed a classification of considered aspects and factors. This was necessary as the studies used over two hundred different aspects and factors. We decided to structure these aspects of UPI and system success and context factors in five categories and additional subcategories. The main categories for aspects of UPI are development process and human aspects. The categories for context factors are system attributes and organizational factors, and one category for aspects of system success. They were defined by a top-down approach. The subcategories were defined by a bottom-up approach from the 231 different research aspects that were named in the studies.

The category *development process* includes all aspects that involve active participation or communication, as well as the roles and responsibilities of the people involved in software development. Various aspects that rather occur on a psychosocial level, such as the attitudes or beliefs of the participants, were combined in the category *human aspects*. Context factors based on the software system were classified in the category *system attributes*. In the same way, the availability of various resources was summarized in *organizational factors*. The last category *system success* comprises various aspects of system success, e.g., user satisfaction, system use, or system quality (Hwang and Thorn 1999). For detailed definitions of each category and subcategory see Table 1.

6.1.2 Examples of Researched Aspects of System Success (RQ 1.1)

System success in software development has been researched in all of the reviewed studies in various ways. Some papers used existing conceptual models, such as the TAM (Davis et al. 1989) with the aspects “Perceived Ease of Use” or “Usefulness”. But not all papers used predefined models. Various terms have been found for “User Satisfaction”, which is the aspect used most often. Also “Project in Time and Budget” has been used in various ways, e.g., “Process Satisfaction” or simply “Project Success”. In Table 2 examples of originally studied aspects are presented. Table 7 in the Appendix, lists which paper studied which aspect of system success.

After the data extraction (researched aspects of UPI and system success, context factors, significant correlations between two aspects, and amount of participants), we classified the 231 different aspects and context factors of the papers in our subcategories and categories. We then counted the amount of unique studies for each category and subcategory, which gives us an answer to research question 1.1 (see Fig. 5). As one study can examine several subcategories within a category, the sum of unique papers for all subcategories is not the same as the value of unique papers for that category.

6.1.3 Correlations of Aspects of UPI and Context Factors to System Success (RQ 1.2,1.3, and 1.4)

For each correlation between two categories, we counted the amount of unique studies. This helped to answers the research questions:

RQ 1.2 Which correlations between these aspects have been studied?

Table 1 Definitions of categories and subcategories

Aspects/Factors	Category/ Subcategory	Definition
Aspects of UPI	**Development Process**	**Different activities of project participants (i.e., end users, developer) that contribute to the system development**
	User Participation	Behavior and activities users perform in system development process, e.g., being the leader of the project team, having responsibility for the overall success of the system, and being responsible for selecting hardware or software, estimating costs, requesting funds (Hartwick and Barki 1994)
	User-Developer Communication	Communication, evaluation, and approval activities that take place between users and IS staff (Hartwick and Barki 1994), also the frequency, content and direction of that communication
	Mode of Development	Depending on which roles are mainly responsible for development, the development process can vary, e.g., the system can be developed by developers, by end-users directly or in a cooperative way between these groups (Zeffane et al. 1998).
	Human Aspects	**Attitude or beliefs of project participants**
	User Involvement	Psychological state of the individual, and defined as the importance and personal relevance of a system to a user (Hartwick and Barki 1994). Also refers to the degree of users' perception on their sense of ownership toward the system (Wu et al. 2006)
	User's Motivation	Motivation is a rationally calculative perspective that an individual's involvement in an activity arises from his/her desire to obtain rewards, including the instrumentality of creating opportunity and improving conditions of work (Chang et al. 2010)
	User's Intention to Use	A function of attitudes towards a behavior and subjective norms (i.e., influence of people in one's social environment) that has been found to predict actual behavior (Hunton and Beeler 1997)
	User's Attitude towards System	Affective or evaluative judgment of the user towards the system, i.e., the extent to which s/he feels the system is evaluated good or bad (Barki and Hartwick 1994; Lin and Shao 2000)
	User's Ability in IT Projects	Ability that enables the users to participate as a member of the systems development team and accomplish the goals of project, e.g., previous experience (Chang et al. 2010)
	User's Beliefs about Developers	Attitude and beliefs of users regarding the behavior of developers, e.g., whether they take the users seriously (Amoako-Gyampah and White 1993) and whether the decision process is fair (Harris and Weistroffer 2009; Hunton and Beeler 1997)
	Developer's Attitude towards User	Attitude of the systems developers towards the users, e.g., are they treated with dignity and do they get informed (Amoako-Gyampah and White 1993; Gefen et al. 2008)
	Disagreement/ Conflict	Divergence of opinions and goals that can lead to conflicts, as well as their resolution possibility (Barki and Hartwick 1994)
Context factors	**System Attributes**	**Attribute or challenge of the to-be-developed system**
	Complexity	Complexity of the actual organizational task(s) being supported by the systems project under study (McKeen and Guimaraes 1997) as well as the ambiguity and uncertainty that surrounds development of that system (Lin and Shao 2000)

Table 1 (continued)

Aspects/Factors	Category/ Subcategory	Definition
	Uncertainty	Extent of stability of business environment and management and resulting conflicting requirements (El Emam et al. 1996; McKeen et al. 1994)
	Organizational Factors	**Influences on the project that come from the organizational context of the IT projects**
	Top Management Support	Support through recognition and fast decisions of high level managers (Rouibah et al. 2008)
	Organizational or Managerial Culture	Shared mental assumptions that guide actions in organizations, e.g., harmony-oriented or control-oriented culture (Bai and Cheng 2010); management style can be distinguished between people- or task-oriented (Lu and Wang 1997)
	Availability of Resources	Existence of project resources, such as system plan, project mission and goals, and training (Amoako-Gyampah and White 1993)
Aspects of system success	**System Success**	**Assessment, whether the IT project and the resulting system has achieved its objectives.**
	User Satisfaction	User's degree of favorableness with respect to the system and the mechanics of interaction (Wixom and Todd 2005)
	Ease of Use	Degree to which a user expects the target system to be free of effort (Amoako-Gyampah 2007), also refers to system friendliness and handling in system's use (Wixom and Todd 2005)
	System Use	Frequency of use of the developed system (Hartwick and Barki 1994)
	System Quality	Structured set of characteristics such as functional suitability, reliability, usability, performance efficiency, compatibility, security, maintainability and portability of a system (ISO 9126–1, 2011)
	Data Quality	Degree to which the characteristics of data satisfy stated and implied needs when used under specified conditions (ISO 25012), e.g., accuracy, consistency and availability of data within the system (Zeffane et al. 1998)
	Project in Time and Budget	Project efficiency and effectiveness in terms of schedule, budget, and work quality (Chang et al. 2010)

Table 2 Overview of aspects of system success

Subcategory	Examples of original aspects
User satisfaction	End-User Computing Satisfaction, End-User Satisfaction, Information Satisfaction, Outcome Satisfaction , Perceived System Usefulness , Perceived Usefulness, System Acceptance, System Satisfaction, Usefulness, User Assessment, User Information Satisfaction, User Satisfaction
Ease of use	Perceived Impact on Work, System Friendliness
System use	Intention to Use, System Impact, System Usage, Time Spend Using
System quality	Accessibility, Accuracy, Completeness, Flexibility, Perceived System Quality, Product Success
Data quality	Appropriateness of Format, Availability of Historical Data, Data Accuracy, Data Consistency, Data Sufficiency
Project in time and budget	MIS Project Success, Overall Success, Process Satisfaction, Project Completion, Project Performance, Project Success, Successful Implementation

RQ 1.3 What are the characteristics of those correlations (percentage of studies reporting positive or negative correlations, variation)?

RQ 1.4 Which further evidence on context factors and their correlations to the aspects of UPI and system success is reported?

An overview of the meta-analysis of SEMs is given in Fig. 5. Each box represents a category with the corresponding subcategories in bullets. After each (sub)category the number of unique studies is stated in brackets. Each correlation is depicted as an arrow and labeled with # studies where # = is the amount of studies that considered that correlation. Each correlation between two categories is labeled with a number in a circle. In addition to the amount of studies, the split between positive and negative studies and the variation of the correlations is specified in Table 3.

Findings on the Category Level **Categories:** Regarding the aspects of UPI, most of the 86 studies examine aspects of the categories development process (71 %), human aspects (49 %) and system success (87 %), thus investigat correlations among these categories. The context factors system attributes and organizational factors only played a minor role in empirical research (less than a dozen studies). Furthermore, it is noticeable that user participation (75 % of all development process aspects), user involvement (49 % of all human aspects) are the most researched subcategories, and user satisfaction is the most common success factor with 51 % of all aspects.

Correlations between categories: In general, the vast majority of studies show positive correlations of aspects of UPI to system success. Only 10 % of the correlations are negative (14 of 146 links). This is in line with most authors such as (Harris and Weistroffer 2009;

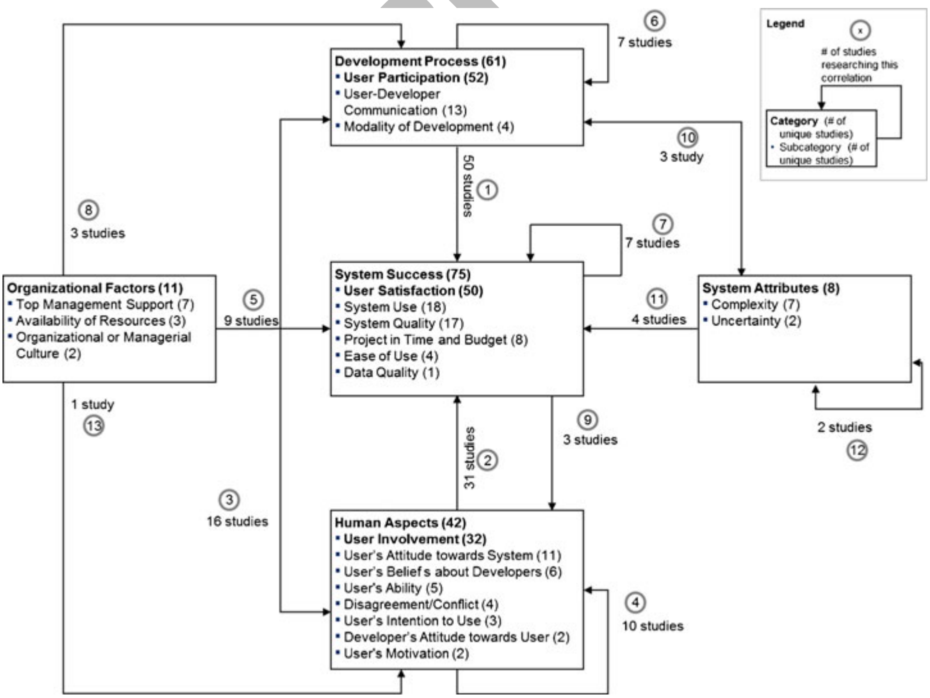

Fig. 5 Overview structural equitation model (SEM) of surveys and meta-studies

Table 3 Amount of positive and negative studies and variation

Link	# Positive studies	# Negative studies	Variation of correlations
1	47	3	−0,47–0,69
2	27	4	−0,18–0,64
3	13	3	−0,97–0,93
4	10	0	0,03–0,75
5	9	0	0,04–0,57
6	7	0	0,27–0,85
7	7	0	0,06–0,85
8	3	0	0,17–0,36
9	3	0	0,24–0,93
10	3	0	0,17–0,36
11	1	3	−0,30–0,28
12	2	0	0,53–0,53
13	0	1	−0,22–−0,22
Sum	**132 (90 %)**	**14 (10 %)**	**−0,97–0,93**

Hendry 2008; Hope and Amdahl 2011; McGill and Klobas 2008). In the following, we explain the labels for the correlations in Fig. 5.

Considering the correlations, the most researched link (50 studies) is the one from the development process category to the system success category (1). However, three of them described negative correlations. The second most studied link is the one between human aspects and system success (31 studies) (2). Even though 87 % show a positive relationship, four studies have measured a statistically significant negative correlation. Not surprisingly, there are also a couple of studies that look into the link between development processes and human aspects (3). This research has been started by (Hartwick and Barki 1994) who distinguish between the definition of user participation as an active part the user performs in software development and user involvement as the more cognitive part in the cooperation with the end users. Understanding the cognitive aspects of cooperation with the user and the correlation between these aspects was in the interest of seven studies (4). Even though all of them showed a positive correlation, some show small values indicating a very low correlation, with the range starting with 0.03. The effect of organizational factors such as top management support as a context factor for system success was examined by nine studies, which revealed positive correlations from 0.04 to 0.57 (5). It is interesting, that seven studies looked into the correlation between the development process aspects, e.g., between user participation and user-developer communication (6). Moreover interdependencies between the various system success factors were looked into; some only depend slightly on each other (0.06 significant correlation), but others quite strongly with a correlation of 0.85 (7). Three studies did not directly focus on the effect of organization factors on system success, but rather on the effect of organizational factors on development process aspects (8). For example: do projects with higher top management support have higher user participation and/or is there more user-developer communication? An interesting correlation is from the system success category to human aspects (9). Normally one would assume that all studies considered the effects of human aspects towards system success. However, three studies also wanted to figure out, which dependencies exist the other way round. The context factor system attributes have actually been studied more thoroughly with regard to negative effects on the development process category. Most people would naturally agree that it is harder to develop a more complex and uncertain system (10). Therefore, these three studies wanted to figure out, if this has an effect on user participation or user-developer communication. One study showed a negative effect from system attributes towards system success factors (11). Given the fact, that it is harder to implement a more complex and uncertain system, it is also harder to lead it to success. Therefore, this negative link is easy to understand. Furthermore, two studies looked into the connection between uncertainty and complexity (12). Lastly, there is one study that showed a negative link from organizational factors to human aspects (13).

6.1.4 Overview of Positive and Negative Studies (RQ 1.3 and 1.4)

In order to study the overall positive effect of UPI on system success, we separated the positive and negative studies (see Table 3). We structured them based on correlations of one category (category 1) to another category (category 2) in the Appendix in Table 8. As mentioned above, most studies show positive results from aspects of UPI towards system success. We defined a study as negative if it reported one negative correlation. That means, if one study tested two correlations of development processes (e.g., user participation and user-developer communication) with aspects of system success and one of them is positive and the other is negative, the study was counted as a negative study. We wanted to prevent any concealment of negative results, also known as the publication bias (Kitchenham and Charters 2007). When we reference a study from a meta-study, we name the meta-study reference and the original reference in brackets.

The first negative study (Barki and Hartwick 1994) showed that an increase in user participation can correlate negatively with the possibility of conflict and disagreement within the project team. This is reasonable as an increase in active cooperation between users and developers will also have a higher potential for conflicts. From the two studies covered in the meta-survey (Cavaye 1995) the study of Kim and Lee 1986, showed a negative link from user participation to the users' attitude towards the system, and Robey and Farrow 1982, were cited with two negative correlations of user participation to perceived influence and conflict resolution. Both negative correlations can be explained by the fact that an increased participation can also lead to higher expectations of the users.

Another study cited by (Cavaye 1995) of Tait and Vessey 1988, indicated a negative correlation of user participation with user satisfaction. Zeffane et al. 1998 covered by the meta-survey (Harris and Weistroffer 2009), showed a negative relationship of the aspect mode of development to data quality. They figured out that depending on who has the main responsibility for development (e.g., the end user or developers) the data quality could be influenced negatively. This might be explained by missing technical competence of the end users. In Kujala's meta-study, Heinbokel et al. 1996, reported four different negative correlations of user participation with various factors that we clustered on the aspect project in time and budget. This is reasonable as participation binds resources of a project.

Moreover, four studies associated human aspects with negative system success. Two studies from (Amoako-Gyampah and White 1993; Amoako-Gyampah 2007) revealed a negative correlation of the developer's attitude towards the user and of the user's intention to use with system success. Doll and Torkzadeh 1991, covered by (McGill and Klobas 2008), describe a negative link from the desired involvement of the users to user satisfaction, but the correlation is very low with 0.03. In addition to their previously described negative correlation, Zeffane et al. 1998, covered by the meta-study (Harris and Weistroffer 2009) stated that human aspects such as involvement in the functional design or the system definition have a negative effect on system success factors such as data quality. This might be also due to missing technical knowledge of the user.

(Amoako-Gyampah and White 1993) figured out that the context factor availability of resources, specifically here the availability of a project plan, has a negative influence on user involvement. This can possibly be explained by the fact that such a plan does not encourage flexible involvement and therefore does not improve the user's psychological state towards the system. The last three negative studies (El Emam et al. 1996; Palanisamy and Sushil 2001; Yetton et al. 2000) of system attributes towards system success are expected to be negative. It is easy to understand that complexity and uncertainty of a system will rather prevent system success than increase its probability.

6.1.5 Amount of Participants in Correlations of Aspects of UPI and Context Factors with System Success (RQ 1.3 and RQ 1.4)

As described above, another characteristic analyzed in the correlations is the amount of participants involved in each study (RQ 1.3). This helps to increase the credibility that increased UPI leads to increased system success. In order to inspect the correlation on a subcategory level, we used the cumulative amount of participants in studies as the relevant factor. We argue that a significant correlation validated by more participants is of higher credibility and also indicates a higher research interest.

As the amount of correlations between each pair of subcategories would be too high for this report, we identified seven links where more than 1000 participants were asked during the studies. Only one study of these seven links shows a negative correlation, therefore we did not separate positive and negative studies. An overview of the links can be found in Fig. 6.

Findings on a Subcategory Level In the following, we will explain the links A–G in Fig. 6. Regarding the measure "cumulative amount of participants", the relationship between user involvement and user satisfaction is the one with the highest correlation credibility (A). We can conclude that the interference of the individual's psychological state defined as the importance and personal relevance of a system to a user (Hartwick and Barki 1994) with his or her satisfaction is quite relevant for researchers. As 3,980 participants in the study stated positive results, this is also a strong argument for RQ 1. In addition, the correlation between user participation and user satisfaction has been researched intensively (4,476 participants) (B). As none of the studies showed a negative correlation, we have evidence that an increase of user participation leads to a higher satisfaction of the user with the system. The dependency between user satisfaction and system use was the focus of four studies and 1604 participants have been studied regarding that correlation (C). In addition, if the system is easier to use (ease of use) users are more satisfied (D).

Furthermore, 1311 participants were asked and revealed solely positive correlations between user involvement and system use (E). A lot of research has been done studying the interdependencies to achieve user satisfaction (F). For example, correlations between system benefits perceived by users or information satisfaction and user satisfaction have been researched. Also the question of what is relevant in order to understand user participation is a research focus, e.g., Barki and Hartwick's separation in hands-on activities and overall responsibility (G).

6.2 Results of the Methods Papers

We wanted to figure out the *characteristics of the methods which aim to increase UPI in software development (RQ 2)*. Therefore, we used a variation of the line of argument synthesis

(Kitchenham and Charters 2007) by first analyzing the individual papers with regards to their targeted issue, the validation context, and their proposed solution. We identified 36 papers that describe methods of UPI (RQ 2.1). Afterwards, we made an attempt to analyze the set of studies as a whole. We analyzed which activities in software development are affected by these methods (RQ 2.2). In order to compare them to the surveys and meta-studies, we identified which aspects of UPI and system success as well as context factors (on a category and subcategory level) are influenced and targeted (RQ 2.3). In order to get some background of the existing research, we analyzed in which context (development methods, industry and software system) these methods have been validated (RQ 2.4). Lastly, we derived an overview of practices including examples of methods and ordered them by the activities in software development (RQ 2.4).

6.2.1 Methods to Increase UPI in Software Development Projects (RQ 2.1)

Overall, we selected 36 papers with methods to increase UPI in software development (see summaries of the papers in the Appendix in Table 10). This shows a broad variety in this research area. A list of all papers can be found in the Appendix in Table 6.

6.2.2 Software Development Activities Affected by the Methods (RQ 2.2)

With the intention to determine the variety and breadth of the different methods, we did a first analysis of the activities in software development which are mainly affected. We used the activities based on (Sommerville 2007), who suggests that general activities of all software processes are software specification, software design and implementation, software validation and verification, and software evolution. In addition, planning and project

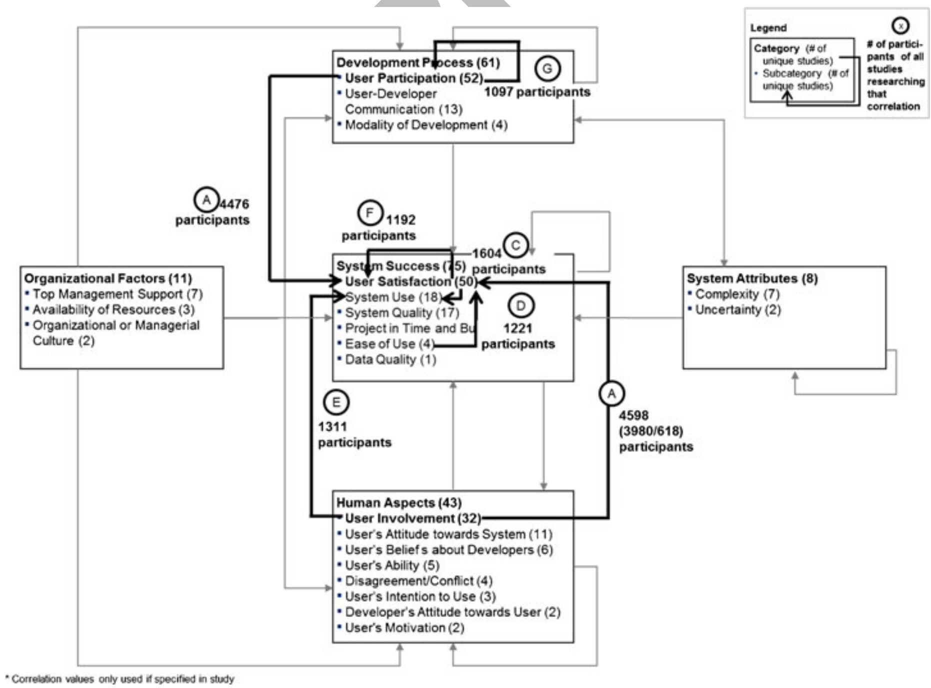

Fig. 6 Overview of links with most participants on subcategory level

management is a critical activity, as software development is always subject to budget and schedule constraints which are set by the organization developing the software.

The overview of the methods studies structured by the different activities in software development can be seen in Fig. 7.

Most papers (33 %) consider all or several activities of software development for UPI. In addition, 11 % of the methods are influencing the planning and project management setup. Almost a quarter (19 %) of the methods focus on the early steps of software development (i.e., SW specification & requirements engineering), which is in line with (Kujala 2008) who highly promotes early user involvement. Four studies specifically focus on requirements engineering, and the other three take a broader view in software specification. 11 % focus on the design and implementation activity. To our surprise, only one paper (3 %) focuses on the verification and validation activity. Lastly, 22 % of the papers focus on the software evolution activity.

6.2.3 Targeted Aspects of UPI, System Success and Context Factors (RQ 2.3)

As we analyze quantitative studies (surveys and meta-studies) and qualitative studies (methods papers) in our systematic mapping study, we also integrated the results of those two branches. Therefore, we analyzed which aspects of UPI, system success, as well as context factors were targeted and influenced by the methods used. An overview of which methods paper target which category and subcategory is given in the Appendix in Fig. 9.

Development process (94 % of all studies), human aspects (81 %) and system success (100 %) are the categories most influenced by the methods that were used. On a subcategory level, user-developer communication seams very important as 69 % of the studies did target

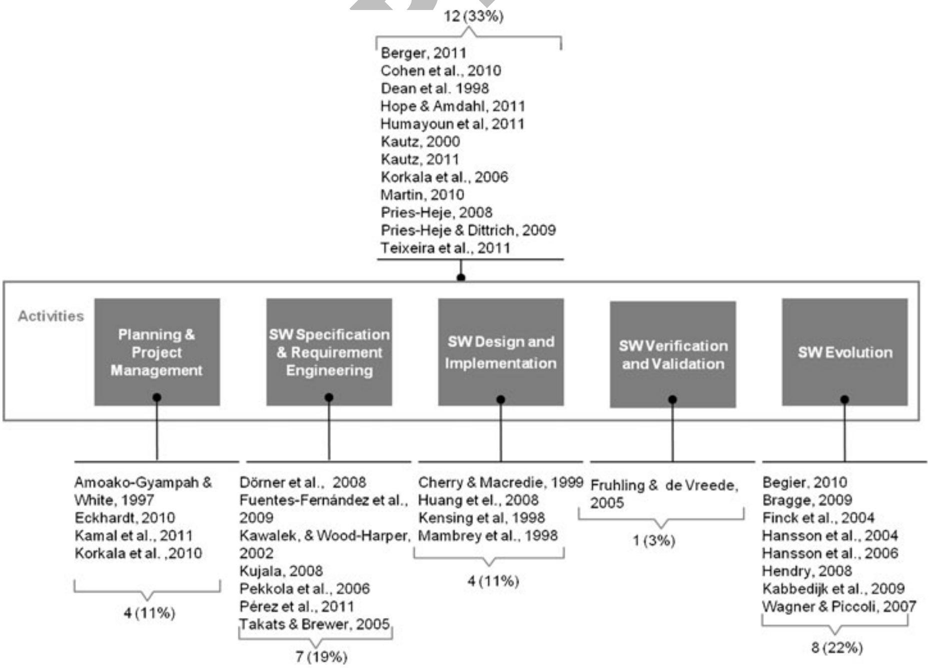

Fig. 7 Overview of methods studies classified by affected software development activities

or influence this subject, whereas only 23 % of the surveys studied the influence of user-developer communication. Also, the mode of development is a very important aspect as over a third (31 %) of the methods is influenced by the responsibilities for development. In the human aspects category, user involvement was the subcategory most influenced, but to our surprise the developer's attitude towards the user is the second most targeted subcategory with 39 %. System attributes and organizational factors have not been influenced significantly by the methods used. System success was targeted by all papers, with system quality as the most important target for 92 % of the methods. In addition, system quality is the single goal for 6 of the 36 papers in terms of system success. User satisfaction is the second most target (53 % of the studies), but this system success factor is mostly used in a combination with other factors.

6.2.4 Validation Context of the Methods (RQ 2.4)

With the intent to give researchers an impression in which context most of the studies were validated, we analyzed the validation context for each methods paper. Therefore, we extracted which UPI method or development method was used. In case the paper named a method for UPI, we preferred that method over the development method. Furthermore, we looked at the industry and the software systems in which the proposed method was validated.

The distribution of UPI as well as the development methods can be seen in Fig. 8. Beside the four papers that do not name a clear method, most papers used agile development methods in their validation (31 %). Furthermore, nine papers (25 %) use participatory design methods. In contrast to agile methods are the heavyweight methods, i.e., the waterfall approach, with six papers (17 %). The two UPI methods, user-centered design and participatory design, are similar methods, but mainly differ in their rationale of why to involve users. Participatory design emphasizes democracy and skill enhancement (Kujala 2008). Therefore, users must not only be part of the design process, but actually are involved in the decision making. Other approaches, such as user-centered design, focus mostly on gaining varied information from users. This basically means the user's context is very important for the software system design, but users do not necessarily have a say in the final decisions

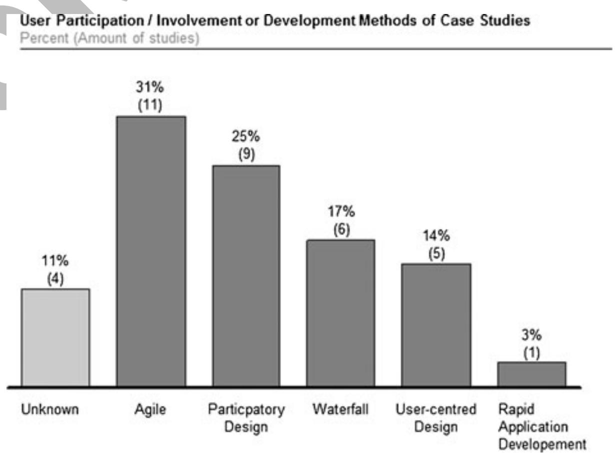

Fig. 8 Development the methods of case studies

(Kujala 2008). Overall, user-centered development methods are used in five case studies. Lastly, one paper uses the rapid application development method where, similar to the agile methods, the focus is on a fast running application with a prototype-like approach (Dean et al. 1998).

Beside the methods used in validation, some other context-related information about the case studies was analyzed.

Interestingly, 15 of the 36 papers (42 %) used a public environment for their validation. This includes case studies in public administration, defense, or educational organizations. Furthermore, only 14 % (5 papers) used more than one case study in different industries.

Regarding the researched software systems for validation in the case studies, the area of information systems was by far the most frequently used one with 72 %. 36 % of the papers did not specify the software system further, but some gave specific system descriptions (17 % enterprise systems, 14 % enterprise resource management systems and 6 % expert systems).

6.2.5 Practices of the Proposed Solutions (RQ 2.4)

Lastly, we analyzed the proposed solutions of the 36 methods papers. We used a reciprocal translation, an approach to qualitative synthesis (Kitchenham and Charters 2007), which can be used if all studies have a similar topic. In this approach, researchers are attempting to provide an additive summary by "translating" each case into each of the other cases. With the intent to make it easier for practitioners and researchers, we structured the practices by the software development activities (planning/project management, SW specification & requirements engineering, SW design & implementation, SW verification & validation, and SW evolution). Furthermore, we distinguished between what needs to be done (practices) and how it is done (examples for UPI).

One can see that there are practices for UPI for every activity; thus, there is a broad variety of methods available in the research community. The planning and project management activity reveals the most frequently used practices and therefore seems to be quite important within this activity. We separated the practices into the three major identified categories (set up of communication structures, set up of project management and set up of project environment). Most of the suggested practices are grouped into the category "set up of communication structures". However, most examples are named for the practice "clarify the project visions based on users needs", which is part of the project management category. We identify only one example for UPI in the method papers that support the setup of the project environment. For the software specification and requirements engineering activity, we identified several practices, but more remarkable is the wide variety of examples of how to include users in that activity of software development. To increase the user participation and involvement in the design and implementation activity, a lot of the identified practices and examples are based on participatory design (e.g., (Cherry and Macredie 1999)) or the agile approaches (e.g., (Kautz 2000)). Therefore, the suggested practices and examples are mostly part of a broader approach throughout the whole software development process. For the verification and validation activity, we only identified four practices and two examples to increase UPI. This is unexpected, as validation is an activity that requires the checking whether a system meets the expectations of the users (Sommerville 2007). Therefore, practices and examples of how to involve users should be more common. In the software evolution activity, we identified several practices and examples to increase UPI. This is an indication that more

Table 4 Practices of proposed solutions from the methods papers

When?/ Activity	What?/Practice for user participation and involvement	How?/Examples for user participation and involvement
Planning & Project Management	Set up of communication structures	
	• Clarify roles of users and mediators to reduce communication barriers, e.g., usability expert, design evaluators, geek interpreter, boundary spanners (Amoako-Gyampah and White 1997; Hope and Amdahl 2011; Eckhardt 2010; Humayoun et al. 2011; Korkala et al. 2010)	• 25 skill set for boundary spanner (Eckhardt 2010) • Role descriptions for customer's apprentice, customer pairing. customer boot camp (Martin et al. 2010)
	• Define a person to play the role of the user upfront and ensure daily communication, if distributed development prevents an on-site customer (Korkala et al. 2010)	
	• Set up asynchronous communication in case face-to-face, is infeasible due to distributed environment	
	• Identify the right users (Amoako-Gyampah and White 1997; Kamal et al. 2011; Kujala 2008)	• Stakeholder analysis (Kamal et al. 2011)
	• Set up reporting structures that involve users and developers (Berger 2011)	
	• Keep people informed and give them timely feedback (Amoako-Gyampah and White 1997; Begier 2010)	• Charts in meeting rooms (Amoako-Gyampah and White 1997) • Email listserv (Korkala et al. 2010)
	• Use shared representations to mediate communication between different professional groups (Pries-Heje and Dittrich 2009)	
	Set up of project management	
	• Ensure project manager's visibility (Amoako-Gyampah and White 1997)	
	• Set up development plan based on the user's need/input and share with users (Dean et al. 1998; Kautz 2000)	• User survey (Dean et al. 1998)
	• Clarify project vision with a high level concept based on users' needs (ideally with users on site) (Cohen et al. 2010; Kensing et al. 1998; Takats and Brewer 2005)	• Workshop with structured agenda using a Group Solve pattern, and produce 1-page description of vision and 1-page logical architecture (Takats and Brewer 2005) • Big Picture Up-Front Workshop (Martin et al. 2010) • Presentation rounds/hearings within different organizational units (Kensing et al. 1998) • Initial analysis of the organization's own documents (Kensing et al. 1998) • Identification of critical success factors (Kensing et al. 1998)
	• Use a globally available project management tool with daily project status for all participants (including users) (Korkala et al. 2010)	

Table 4 (continued)

When?/ Activity	What?/Practice for user participation and involvement	How?/Examples for user participation and involvement
	• Have an ERP competence center to mediate between users and external IT experts (Pries-Heje and Dittrich 2009)	
	Set up of project environment	
	• Use incremental project lifecycle with iterative development (Berger 2011)	
	• Colocate user and developer (Berger 2011; Kautz 2000)	• Joint Application Development (JAD) workshops (Berger 2011)
	• Have IT professionals work for some time in the user organization in order to understand the needs and observe existing practices (Pries-Heje and Dittrich 2009)	
SW Specification & Requirements Engineering	• Visit users in their own environment and explore their needs (Kujala 2008; Martin et al. 2010; Teixeira et al. 2011)	• Contextual and Sociotechnical analysis (Kawalek and Wood-Harper 2002) • Focus group and direct observations (Teixeira et al. 2011)
	• Describe the current situation (Kujala 2008)	• Task hierarchy, scenario and user-needs table within field studies (Kujala 2008) • Strategic analysis (Kensing et al. 1998) • In-depth analysis of selected work domains (Kensing et al. 1998)
	• Identify user-dependent scenarios (Pérez et al. 2011)	
	• Guide user representative from the analysis of business needs to the identification of system requirements (Dean et al. 1998)	
	• Conduct requirements analysis face to face with users (Korkala et al. 2006; Takats and Brewer 2005)	• Paper-based prototypes (Korkala et al. 2006) • Activity theory requirements engineering (Fuentes-Fernández et al. 2009) • Focus groups and card sort methods (Humayoun et al. 2011) • Heavily facilitated workshops with "be visual" and "forced rank" pattern (Takats and Brewer 2005)
	• Use software support to elicit requirements while user is using a prototype	• Infrastructure probes (Dörner et al. 2008) • Domain specific visual language (Pérez et al. 2011) • Multi-methodological information system development approach that utilizes prototyping (Pekkola et al. 2006)
	• Have a thorough requirements specification (for off-the-shelf systems) as a basis for the contract (Pries-Heje 2008)	
SW Design & Implementation	• Involve onsite customers in requirements and story card prioritization in design approval (Kautz 2011)	
	• Take existing practice of users as the starting point for the design process. (Cherry and Macredie 1999)	• Cooperative prototyping (Cherry and Macredie 1999)

Table 4 (continued)

When?/ Activity	What?/Practice for user participation and involvement	How?/Examples for user participation and involvement
	• Let users articulate their requirements through prototypes that can be iteratively modified (Cherry and Macredie 1999) • Allow users to experiment with different work scenarios (Cherry and Macredie 1999)	
	• Have structured brainstorming session to transforms general characteristics into a common design strategy for both users and developers (Cherry and Macredie 1999)	• Future workshop (Cherry and Macredie 1999; Kensing et al. 1998)
	• Develop visions of the overall change in design and anchor the visions by management and the steering committee, technical and organizational implementation team and the users	• Visits to "similar" work places (Kensing et al. 1998) • Design workshops (Kensing et al. 1998)
	• Have designers visiting the workplace and having contact with users (Mambrey et al. 1998)	• Osmosis (interviews, user workshops, active user services, and simply being present at the workplace) (Mambrey et al. 1998)
	• Let skilled spokespersons of users participate in development work (Berger 2011; Hope and Amdahl 2011)	• Dynamic system development methods work (Hope and Amdahl 2011) • Paper prototypes (Humayoun et al. 2011) • User advocacy (Mambrey et al. 1998)
	• Allows consideration of alternative work processes by playing them out and confronting the problems created (Cherry and Macredie 1999)	• Organizational gaming (Cherry and Macredie 1999)
	• Plan content of next iteration with users on site (Korkala et al. 2006)	• Planning games (Kautz 2000) Mobile-D (Korkala et al. 2006)
	• Evaluate design with end users through quick evaluation methods and improve the design based on prototype evaluation (Humayoun et al. 2011)	• Evaluation experiments run from within the development environment with UEMan and TaMUlator tools (Humayoun et al. 2011)
	• Have weekly feedback meetings with onsite customers during working software presentations (Kautz 2011)	•
	• Have iterative design process with task analysis, scenario design, design implementation and usability testing and evaluation (Huang et al. 2008)	• Brainstorming, focus groups, mockups and usability quiz (Huang et al. 2008)
	• Have mid-iteration communication with users (Korkala et al. 2006)	• Face-to-face meetings/videoconferences (Korkala et al. 2006)
	• Develop in iterations and get feedback quickly from the users through the test of the software versions (Teixeira et al. 2011)	• eXtremProgramming (XP) (Teixeira et al. 2011)
	• Allow change requests for the software design from onsite users in weekly feedback loops (Kautz 2000)	• Re-calibration workshops (Martin et al. 2010)
	• Allow necessary customizations to the system (for off-the-shelf systems) (Pries-Heje 2008)	
SW Verification & Validation	• Use prototypes for evaluation with users (Cohen et al. 2010; Dean et al. 1998; Humayoun et al. 2011)	• Heuristic evaluation, question-asking protocol, and performance measurement (Humayoun et al. 2011)

Table 4 (continued)

When?/ Activity	What?/Practice for user participation and involvement	How?/Examples for user participation and involvement
	• Let end users evaluate modules supported by automated tools	
	• Let onsite user representatives collect feedback and proposals for improvements from other users based on the working software (Kautz 2000; Martin et al. 2010)	• "Road shows" from onsite users to other users (Kautz 2011; Kautz 2000; Martin et al. 2010)
	• Prepare and perform acceptance test with onsite customers (Kautz 2011)	
SW Evolution	• Encourage users to suggest new features asynchronously (Bragge 2009)	• Feedback function within system (Bragge 2009) • Mailing lists with active participation of developers (Hendry 2008) • User questionnaires (Begier 2010) • Electronic web interface for feedback and proposals (Hansson et al. 2004)
	• Obtain feedback from users concerning system limitations, faults, and proposals for future development through various channels (Hansson et al. 2006) • Use support as a channel for feedback and change proposals (Hansson et al. 2006) • Keep track of user feedback (Hansson et al. 2006)	• Support calls, user meetings, courses, the website and newsletter (Hansson et al. 2006) • Customer relationship management tools (Hansson et al. 2006)
	• Exchange of information about and feedback for the ongoing development (Finck et al. 2004) • Give users an incentive for expressing problems and ideas about system usage (Finck et al. 2004) • Inform users through a facilitator about design decisions of next release based on requirements from discussion forum (Finck et al. 2004)	• Discussion forum in groupware system (Finck et al. 2004)
	• Set up a synchronous feedback session with user groups (Kabbedijk et al. 2009)	• IT helpdesk (Bragge 2009; Hansson et al. 2004) Virtual group support systems
	• Set up usability workshops with users	• Customer participation sessions with idea feedback and user suggestions (Hansson et al. 2004; Kabbedijk et al. 2009) • Thinklets (Fruhling et al. 2005)
	• Acknowledge that users have limited interest before "go-live" and involve them afterwards (Wagner and Piccoli 2007)	

software projects are being developed continuously and therefore are dependent on user feedback for new releases.

It should also be mentioned that this list of practices is not meant to be applied completely in one project, but should rather be used as a list for selection of different practices. Software engineering projects can pick some activities for their own purposes. The summary of the practices is shown in Table 4.

7 Discussion

In this section we discuss the results of our systematic mapping study. We start with principal findings, discussing the findings based on our research questions on a detailed level and then relating them to the main research questions RQ 1 and 2. Afterwards we discuss the strengths and weaknesses with threats to validity. Finally, we include some further historical analysis of the researched papers to discuss trends of UPI research.

7.1 Principal Findings

User participation and involvement is an important research topic (RQ 1.1 and 1.4). From the fact that 231 different aspects have been dealt with in the 86 studies, we can see that the topic of UPI has been studied on a broad scale in various research areas. The variety of aspects of UPI in the categories Human Aspect and Development Process, the different context factors, and the various system success aspects, show that it is a complex field to measure and that the various influences are difficult to define. In addition, the mere fact that we found 86 studies that researched the effects of UPI on system success shows that the field is important for researchers and practitioners.

Aspects of the UPI have a positive effect on system success (RQ 1.2, 1.3, and 1.4). Given the vast amount of positive correlations, we can conclude that, even though the results are not completely consistent, the amount of studies with positive correlations of the various aspects of UPI on system success provides evidence of a robust and transferable effect (Kitchenham and Charters 2007). Beside the fact that most surveys have researched the effect of aspects from the development process or the human aspects category on system success, it is remarkable how many studies undertook the effort of studying various interdependencies among the other categories or subcategories. User satisfaction seems to be the most appropriate variable to measure system success, but this could also be biased by the researchers, as they have a human focus when studying UPI.

Most studies with negative correlations were published more than 10 years ago (RQ 1.3 and 1.4). We identified only 14 studies that described any negative correlation between aspects of UPI and system success, as well as context factors. Overall, we can see that most of the studies showing negative results are rather old; there is only one study originally published in the last 10 years. In addition some of the negative correlations can be explained through the researched aspect or context factor, e.g., the system attribute complexity is expected to decrease system success. Furthermore, the correlation value of more than half of the remaining studies is under 0.2, which indicates a low correlation. Four of the negative studies report negative correlations between aspects of UPI or context factors, e.g., between development process and human aspects. That means these negative correlations do not influence the positive effects of UPI on system success. Apart from that, we counted studies as negative if they presented only one negative correlation, and most of them also show other positive results.

Large variations of correlations show complexity to measure and study UPI (RQ 1.3 and 1.4) The analysis of the correlation data of 86 studies show a large variation for most links between aspects of UPI, system success, and context factors. This is an indicator that there is still no clear conceptual model to measure effects of UPI. We therefore consider it important to further research the area of UPI and its effects.

User participation and involvement have a positive effect on user satisfaction and system use (RQ 1.3 and 1.4). Overall, the triangle of user involvement, user participation, and user satisfaction is quite dominant in this field of research. Based on the values of the correlations, the correlation between user satisfaction and system use seems to be studied a lot by researchers and therefore relevant for UPI research. An indicator for the broadness of UPI in research is shown by the analysis of participants that were involved to validate the effects on a subcategory level. The fact that more than 1,000 participants from various studies agreed on positive correlations on a subcategory and only one study reported on a negative correlation indicates the confidence of the identified correlations. The analysis showed that users, who participate in software development, are more satisfied with the system. The same is true for users who are more involved. Therefore, we can conclude that UPI has a clear positive effect on user satisfaction. In addition, the analysis showed that more satisfied users use the system more frequently. Therefore, we can conclude that an increase of UPI increases system use, which is a measure for system success. As a positive correlation between ease of use and user satisfaction has been found, we can also conclude, that if a system is easy to use, the users are more satisfied.

Summary of findings for RQ 1. With regard to our first main research question whether increased UPI leads to increased system success, we conclude that this is shown by our meta- analysis. Thus research and work in the area of UPI in software development is beneficial and should be continued. However, the variety of aspects and context factors that we derived from the studies indicate that there is still no common conceptual model to measure and evaluate these effects. Although we did not focus our meta-analysis specifically on the context factors that influence UPI, we can see that they did not play an important role in our identified studies. We therefore think that more research on the influencing factors in specific contexts is required. Furthermore, the large variation of the identified correlations indicates that more sophisticated empirical studies on the effects of UPI would help the research community.

All software development activities are affected by methods (RQ 2.2). The analysis of affected software development activities revealed that lots of methods focus on all activities of software development, which shows an attempt for a comprehensive approach by most researchers. A clearer focus on one activity might help in the implementation of UPI in real software development projects. Even though it is important to involve users early in the process (Taylor and Kujala 2008; Majid et al. 2010), we think that a lot of important decisions are taken when translating user requirements into system requirements, which happens in the design and implementation activity (Abelein and Paech 2012). Contradicting (Majid et al. 2010), only one method focuses on the software verification and validation activity. But one can see from the practices of the proposed solutions that most of the methods focusing on all activities have validation activities as part of their solution. Furthermore, we did not anticipate so many methods that focus on software evolution. One reason might be that most users start to get interested in a software system only when they are really affected, which is normally after the first deployment of the system (Wagner and Piccoli 2007). Furthermore, most software development nowadays is evolutionary development, which explains the high number of meth9ods that affect the SW evolution activity.

Methods for user participation and involvement target the same categories as the surveys, but show differences on a subcategory level (RQ 2.3). At first sight, the comparison of the aspects researched by the surveys and the aspects targeted by the methods reveals similarities. However, a lot more methods focus on communication between user and developer and the responsibilities for development. Furthermore, many of the methods aim to improve the developer's attitude towards the user, the user's attitude towards the system and user's motivation whereas these aspects were investigated not extensively by the surveys. This could be explained by the fact, that it is quite hard to empirically measure attitudes of humans; however, this is an important goal for a method. In line with the results of the surveys, the context factors system attributes and organizational factors were not highly targeted by the methods, but we also did not focus our mapping study on such context factors. In general complexity seems to be more important for system success than uncertainty of the system. Within the organizational factors, the subcategory top management support was targeted by almost a quarter of the method papers (22 %), which shows that convincing managers of UPI methods are important for a successful implementation of a method. We therefore suggest considering empirical validation of that effect. Another interesting finding was that the methods focus on system quality as the measure most used for system success, whereas the surveys focus on user satisfaction. This indicates that authors suggesting new methods still have a functional or rather technical view on system success.

Most methods were validated in a public environment (RQ 2.4). The validation context of the methods was mainly located in agile environments. This is not surprising as the lightweight methods become more widespread in software development and are rather focused on the user, based on principles of the agile manifesto (Beck et al. 2001). In the analysis of validation context, we could discover that a lot of papers used a public environment for their validation. This can be explained as the access to these organizations is easier. Nevertheless, we think it is important to validate new methods within various environments, such as large companies and organizations in the private sector. Another advice for further research is to validate new methods in more than one case study. Not to our surprise, most researched software systems for validation were information systems. This can be explained by the fact that information systems focus on the support of everyday operations of human beings which leads to a higher importance of UPI (Singh and Kotzé 2003).

The practices derived from the solutions of the methods showed a broad variety within all software activities (RQ 2.4). The structured overview of practices with examples for UPIshows that there are suggestions for each software development activity. Most of the practices are grouped in the planning and project management activity. In line with the analysis of targeted aspects, we can see a focus on communication structures. Figuring out who are the right users for involvement, the setup of structures for how and when to communicate with them as well as for keeping them informed and giving them feedback, are all practices suggested by various papers. Nevertheless, beside some role descriptions only few concrete methods are suggested. This is different in the software specifications and requirements engineering activity, where a lot of interesting methods of how to ensure participation exist. This can be explained by the active research field of requirements engineering. Even though only four studies focused on the design and implementation activity, we could extract some suggestions for practices. Most of them keep the development and design content flexible, e.g., through mid-iteration communication or iteration planning with the users. The suggested participation methods have a connection to agile methods, but also completely new approaches, such as the evaluation experiments that run within the development environment, do exist (Humayoun et al. 2011). Lastly, within the software evolution activity, a lot of specific participation methods for collecting feedback from the user, either asynchronously or synchronously, have been identified.

Summary of finding for RQ 2. The analysis of characteristics of methods that aim to increase UPI in software development shows that lots of different approaches have been developed. However, most of them have been evaluated in rather small projects mainly in the public sector area. This might explain why UPI is still not very widespread in large projects and in private companies and organizations. Nevertheless our meta-analysis shows a clear positive effect of UPI on system success, thus those methods are of value for software development projects. Therefore, we believe that further research on methods to increase UPI specifically targeted to specific contexts (e.g., large projects in private companies) is required.

7.2 Strengths and Weaknesses

In this section, we want to point out the strengths and weaknesses of our study. We show the differences to the other identified meta-studies as well as to another systematic literature review. We discuss the benefits of our study and explain threats to validity.

The six meta-studies that we identified in our systematic mapping study had a different focus or approach than our systematic mapping study. The meta-study of (Harris and Weistroffer 2009) did analyze 28 empirical studies, but only summarized them on a descriptive basis and did not quantitatively evaluate the results. (Kujala 2003) also combined qualitative and quantitative data, but focused on the early steps with regard to requirements management. Thus, this paper did not look into the whole software development process. (McKeen et al. 1994) is a good study regarding the various context factors of user participation, but it focuses solely on empirical studies. This is also the case for (Cavaye 1995; Ives and Olson 1984). The authors try to resolve some contradicting results regarding the effect of UPI. Furthermore, all three studies were published almost 20 years ago. (McGill and Klobas 2008)'s meta-study is also a very interesting overview, but has a focus on a user-developed system, a constraint we did not use in this paper.

A clear strength of our systematic mapping study in comparison to the identified meta-studies is the wide range with which we considered the influence of UPI in software development. We did include statistical surveys and meta-studies to increase confidence in the effects of UPI on a quantitative basis. In addition, we complemented it by a description of various methods, which we analyzed and derived practices from. Furthermore, we used a wide range of sources within three different domains and the number of 3,698 hits of our search string is an indicator for the richness of research we chose our studies from. In total, we comprise results of 58 scientific papers within this systematic mapping study, but considering the six meta-studies about the effects of UPI, even more results are used.

In parallel to our work (Bano and Zowghi 2013) conducted a systematic literature review on the relationship between user involvement and system success. They used a similar search string consisting of synonyms for users, involvement and software development and a mixture of different search strategies. With respect to sources, the electronic sources for the information technology domain are identical to our study, but different databases have been used. Furthermore, they used specific sources of management science journals and DBLP publications profiles of highly cited researchers, but did not include specific journals for participatory design and communication in their search. Given these differences in the search strategy as well as different inclusion criteria, the sets of identified papers have some overlap, but also major differences: 41 studies of the 87 studies analyzed in that review are included in our work. In line with our results, that review confirms the positive effects of user involvement on system success and also argues that UPI is a complex phenomenon and difficult to measure. This increases the evidence of our results that UPI has a positive effect on system success, specifically as the reviews analyzed to a certain degree different studies. The goals of that review and our work differ. (Bano and Zowghi 2013) researched the relationship of user involvement to system success with respect to controversial results in the past. They analyzed the historical development of the relationship and studied the differences in characteristics of the existing evidences. Their finding of an increase in positive effects in more recent decades is interesting evidence that extends our results. Furthermore, (Bano and Zowghi 2013) focus on analyzing current knowledge on the relationship between user involvement and system success. We focus on a deeper understanding of the aspects considered in existing evidence, following a meta-analytical approach. We seek to understand existing practices of methods in order to understand the current research status and to enable new methods to increase UPI.

We also want to point out possible threats to validity of our study. First the selection process was mainly conducted by the first author of the paper, who was a first year PhD student at that time. Due to the large amount of hits from our search string, the initial round of paper selection was only based on title and abstract. The decision which papers are relevant was solely made by the first author. This may indicate a certain degree of subjectivity and therefore is a threat to internal validity. However, as we retrieved such a large amount of duplicates from the different sources, we are convinced that the selection was consistent with the defined criteria. The fact that the first author was a first year PhD student, who just started his research in UPI, ensured that the selection was not biased by pre-known authors. For the following selection, we defined clear exclusion criteria and stated the reason for exclusion in a protocol. The second author then checked the protocol on a random sample base (about 10 % of the results) for validity. The check was done based on the reason for inclusion or exclusion and a check of the paper itself. By following the approach of (Kitchenham and Charters 2007) with a strict process and clear inclusion and exclusion criteria and by the validation check of the second author, we hope to reduce the researcher's bias in the study.

Another possible systematic bias is that authors used other terms for UPI, which we did not cover in our search string. We tried to reduce that risk by using a lot of synonyms and included the common term "participatory design" as an alternative, however it is still possible that we missed some interesting studies. In addition, the topic UPI in software development is not a mainstream research topic, thus it might be possible that some publications appear in places not covered by our sources. Especially the IS community therefore suggests, backward and forward snowballing instead of a search string as search strategy (Jalali and Wohlin 2012). Instead, we tried to overcome this by including specific sources (Participatory Design Conference proceedings, Information Technology & People Journal and the Scandinavian Journal of Information System). Furthermore we used some part of backward snowballing, as we did reference searches for the paper (Al-Rawas and Easterbrook 1996) and the six meta-studies.

Another possible weakness of our approach could be the exclusion criteria “Out of date” (M5). With this criterion, we excluded methods papers, which were published more than 15 years ago. However, we used the research of survey papers that were published before. Although this approach might seem inconsistent, we argue that software and development processes have changed significantly and thus methods, developed more than 15 years ago are not relevant for this study. This can be true for survey papers as well, but we think that here, the statistical correlations are more general and should therefore be used.

7.3 Further Analysis

A historical analysis of all 58 papers plus the 66 papers that were references of the meta study shows, that the topic of UPI has been constantly studied over the last five decades and the first paper was even published as early as 1959 (see Fig. 9). We can see an increase in published surveys until 1997 with some dips in the 80ties. From 1997 onwards, we also included the method papers (method papers published prior to 1997 were excluded based on the exclusion criteria M5). Nevertheless, there was a clear decrease in the 2000 years until 2005. Since then, the research interest on UPI has increased again. Furthermore, one can see, that especially a lot of method papers have been published in the last 5 years. In line with our findings based on our research questions, this analysis shows that even though the topic of UPI in software development has been considered for such a long time, there still is no clear solution of how to implement UPI in practice. Therefore, we believe research specifically on methods for UPI in different contexts is required.

Furthermore, we wanted to identify trends of terminology of the research area of UPI (Table 5). We can see that user participation and user involvement have been mentioned in about the same amount of titles of papers. We therefore think our approach of combining them to the term UPI is useful. However, user involvement was more prominent in the early nineties, whereas user participation has become more

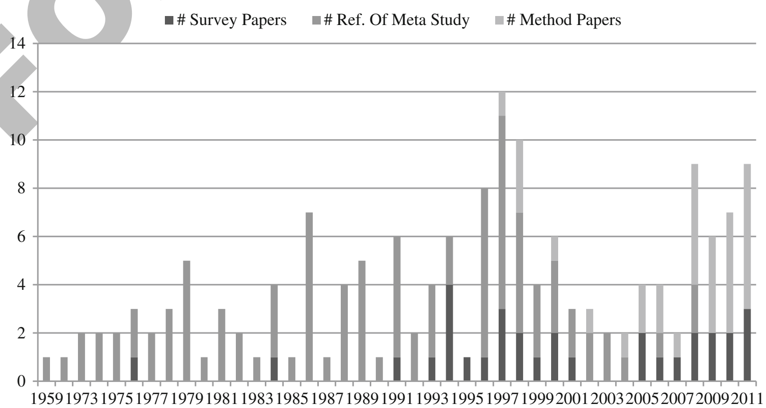

Fig. 9 Historical analysis of the amount of surveys and method papers

Table 5 Historical analysis of important terms

Year	1959	1972	1973	1974	1975	1976	1977	1978	1979	1980	1981	1982	1983	1984	1985	1986	1987	1988	1989	1990	1991
User participation						1									1	1					2
User involvement				1					3		2	2		1		5		3	2	1	3
User satisfaction						1										3		1	3		
Collaboration																					
Communication								1						2							
Year	1992	1993	1994	1995	1996	1997	1998	1999	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	**Sum**
User participation	1		5	1	5	8	3	1	3	1	2		1		3	1	4	1	1	3	**49**
User involvement	2	3	1			2	4	1	2	1	1			2		1	2	2		1	**48**
User satisfaction		3	2		1									1			1	1			**17**
Collaboration										1				1			1	1	1		**5**
Communication			1				1							1	1				3		**10**

popular from 1994 until today. This could also be an indication that active participation of users has been proven to be more effective. User satisfaction, the most common aspect of system success, has not been mentioned in many titles. This indicates it is a commonly used measure for system success, but not an important term in the area of UPI. Especially from 1997 onwards, when we started to include the method papers, communication and collaboration appeared in the titles. This is in line with our analysis of practices from the method papers, and strengthens our assumption that UPI depends a lot on communication between the involved stakeholders.

8 Conclusion and Future Work

In this paper, we describe a systematic mapping study that examines the influence of UPI on system success. We followed the guidelines of (Kitchenham and Charters 2007): defined our research question, conducted a structured identification of research based on a search string, defined clear inclusion and exclusion criteria, and analyzed the resulting 58 papers with regard to our research questions.

The objective of the study was twofold. First, we wanted to figure out if an increase of UPI increases system success. Second, we wanted to identify the characteristics of methods that increase UPI within software development.

To validate the effect of UPI, we used meta-analytical techniques. We extracted the researched aspects, correlation data, variation, and number of participants for validation from the 86 studies. The most important finding is that the vast majority of the derived correlations showed a positive effect, thus we can conclude that *aspects of the development process and human aspects have a positive effect on system success* (RQ 1.2, 1.3, and 1.4)**.** We looked into the 14 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, we found that *most studies with negative correlations were published more than 10 years ago* (RQ 1.3 and 1.4)**.** These results increase the confidence that UPI is beneficial to system success, which is an important finding for other researchers that develop methods to increase UPI in software development. The amount of participants that were involved in the studies indicates the breath and profoundness of this research area. It shows that *user participation and involvement has a positive effect on user satisfaction and system use* (RQ 1.3 and 1.4)**.** Nevertheless, the *large variation of correlations shows the complexity of measuring and studying UPI* (RQ 1.3 and 1.4). Another important contribution of this paper is the developed classification of the aspects of UPI. The main categories of the classification are: 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 (RQ 1.1 and 1.4). This classification can support other researchers interested in studying the aspects of UPI. It could also be used as a starting point to develop a common conceptual model for aspects of UPI and system success, as well as context factors.

From the 36 methods papers, we first analyzed their targeted issues, their validation contexts and their proposed solutions (for a summary of each paper see Appendix Table 10) (RQ 2.1). An important finding is that *all software development activities* (planning & project management, SW specification & requirements engineering, SW design & implementation, SW verification & validation, and SW evolution) *are affected* *by methods*, but only few methods focus on the design and implementation activity (RQ 2.2). This insight can support other researchers in the identification of existing research gaps for methods that aim to increase UPI. In addition, an important contribution of this paper is the structured overview of practices with method examples. The overview shows that *practices derived from the solutions have a wide variety in all software activities* (RQ 2.4)**.** Most of the practices are grouped into the planning and project management activity. This is important as it comprises all activities that are required for the entire project. In line with the analysis of targeted aspects, we have seen a focus on communication structures. The overview is particularly helpful for practitioners, who want to use existing practices and methods to increase UPI in software development. In addition, it can also be valuable to other researchers to understand the state-of-the-art research of UPI methods in software development. The comparison between aspects researched by the surveys and the targeted aspects from the methods reveals that *methods for user participation and involvement target similar categories as the surveys* (RQ 2.3)**.** However, 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* (RQ 2.4). Therefore, we encourage other researchers to validate new methods in private organizations.

Overall, we conclude that the systematic mapping study shows a positive correlation of various aspects of UPI on system success. However, there is still no common conceptual model to measure and validate this effect. Even though it was not the focus of our analysis, the studied papers only show little attention to the influencing context factors. Therefore, more empirical research on aspects of UPI in specific contexts and the various context factors e.g., top management support, should be done. We identified a broad variety of methods to increase UPI in software development, but they have been validated mostly in smaller projects and in the public sector areas. We therefore suggest to further research and develop new methods for other contexts. Especially in large-scale information technology projects, UPI is not a common practice (Alleman 2002). We already sketched elements of a method aiming to enhance user-developer communication in large-scale information technology projects (Abelein and Paech 2012). This proposal was mainly based on personal experiences from the industry context. Given the results of this systematic mapping study, we have further evidence that an increase in UPI, and in particular in user-developer communication, can increase system success. Furthermore, the analysis of aspects did indicate only little focus on organizational factors or system attributes. However, when we consider large information technology projects within big companies, these projects are heavily influenced by factors such as the complexity of the system and the managerial culture of the organization. Therefore, it might be necessary to emphasize those aspects in our new method. In addition, the study reveals that only few methods focus on UPI in the software design and implementation activity, even though within these activities a lot of important decisions are made. Given the fact that many of the methods target the users' motivation, this aspect should also be included in the method design. In line with (Ives and Olson 1984), we think that user satisfaction is a critical factor that leads to higher system acceptance and use. A rise in usage will also increase the value of a system. Therefore, we think a new method should target user satisfaction as an equally important success goal as system quality. In our future work, we therefore want to extend our method proposal (Abelein and Paech 2012). We want to build upon the identified practices and examples we derived from the studied methods.

Appendix

Table 6 List of selected survey, meta study and method papers

Survey and meta studies	Method papers
(Amoako-Gyampah 2007)	(Amoako-Gyampah and White 1997)
(Bai and Cheng 2010).	(Begier 2010)
(Barki and Hartwick 1994).	(Berger 2011)
(Cavaye 1995)	(Bragge 2009)
(Chang et al. 2010)	(Cherry and Macredie 1999)
(El Emam et al. 1996)	(Cohen et al. 2010)
(Gefen et al. 2008)	(Dean et al. 1998)
(Harris and Weistroffer 2009)	(Dörner et al. 2008)
(Hartwick and Barki 2001)	(Eckhardt 2010)
(Hartwick and Barki 1997)	(Finck et al. 2004)
(Igbaria and Guimaraes 1994)	(Fruhling et al. 2005)
(Iivari et al. 2011)	(Fuentes-Fernández et al. 2009)
(Ives and Olson 1984)	(Hansson et al. 2004)
(Kanungo and Bagchi 2000)	(Hansson et al. 2006)
(Kristensson et al. 2011)	(Hendry 2008)
(Kujala 2003)	(Hope and Amdahl 2011)
(Kujala et al. 2005)	(Huang et al. 2008)
(McGill and Klobas 2008)	(Humayoun et al. 2011)
(McKeen et al. 1994)	(Kabbedijk et al. 2009)
(Rouibah et al. 2008)	(Kamal et al. 2011)
(Subramanyam et al. 2010)	(Kautz 2000)
(Wixom and Todd 2005)	(Kautz 2011)
	(Kawalek and Wood-Harper 2002)
	(Kensing et al. 1998)
	(Korkala et al. 2006)
	(Korkala et al. 2010)
	(Kujala 2008)
	(Martin et al. 2010)
	(Mambrey et al. 1998)
	(Pekkola et al. 2006)
	(Pérez et al. 2011)
	(Pries-Heje 2008)
	(Pries-Heje and Dittrich 2009)
	(Takats and Brewer 2005)
	(Teixeira et al. 2011)
	(Wagner and Piccoli 2007)

Table 7 Overview of researched aspects of system success with sources

Subcategory	Source
User satisfaction	Amoako-Gyampah 2007
	Cavaye 1995 (Allingham and O'Connor 1992, Baronas and Louis 1988; Baroudi et al. 1986; DeBrabander and Thiers 1984; Doll and Torzadeh 1989; Franz and Robey 1986; Hirschheim 1985; Kappelmann and McLean 1991; Tait and Vessey 1988)
	El Emam et al. 1996
	Gefen et al. 2008
	Harris and Weistroffer 2009 (Blili et al. 1998; Choe 1996; Doll and Deng, 2001; Guimaraes and Igbaria 1997; Guimaraes et al. 2003; Hsu et al. 2008; Hunton and Price 1997; Lawrence et al. 2002; Lin and Shao 2000; Lu and Wang 1997; Saleem 1996; Santhanam et al. 2000; Yoon et al. 1998)
	Igbaria and Guimaraes 1994
	Ives and Olson 1984 (Edstrom 1977; Gallagher 1974; Guthrie 1972; Kaiser and Srinivasan 1980; Maish 1979; Swanson 1974)
	McGill and Klobas 2008 (Amoako-Gyampah and White 1993; McKeen and Guimaraes 1997; Doll and Torkzadeh 1988; Doll and Torkzadeh 1991; Hartwick and Barki 1994; Hawk 1993; Lawrence and Low 1993; Seddon and Kiew 1996; Torkzadeh and Doll 1999; Torkzadeh and Lee 2003)
	McKeen et al. 1994 (Kappelman and McLean 1991; Olson and Ives 1981; Powers and Dickson 1973)
	Rouibah et al. 2008
	Wixom and Todd 2005
System use	Amoako-Gyampah 2007
	Bai and Cheng 2010
	Cavaye 1995 (Baroudi et al. 1986; Jarvenpaa and Ives 1991; Kim and Lee 1986)
	Harris and Weistroffer 2009 (Choe 1996; Hunton and Price 1997; Lynch and Gregor 2004; Wu et al. 2006)
	Igbaria and Guimaraes 1994
	Ives and Olson 1984 (Lucas 1975; Swanson 1974)
	Kanungo and Bagchi 2000
	Kujala 2003 (Barki and Hartwick 1991)
	McGill and Klobas 2008 (Barki and Hartwick 1991; Hartwick and Barki 1994)
	Rouibah et al. 2008
	Wixom and Todd 2005
System quality	Harris and Weistroffer 2009 (Butler and Fitzgerald 1997; Discenza et al. 2008; Doll and Deng 2001; Foster and Franz 1999; Guimaraes and Igbaria 1997; Kirsch and Beath 1996; Santhanam et al. 2000; Yoon et al. 1998; Zeffane et al. 1998)
	Ives and Olson 1984 (Boland 1978; Gallagher 1974)
	Kujala et al. 2005
	McGill and Klobas 2008
	McKeen et al. 1994 (Franz 1979; Olson and Ives 1981)
	Wixom and Todd 2005
Project in time and budget	Chang et al. 2010
	Harris and Weistroffer 2009 (Jiang et al. 2002; Wu et al. 2006; Yetton et al. 2000)
	Kujala et al. 2005

Table 7 (continued)

Subcategory	Source
	Kujala 2003 (Heinbokel et al. 1996) McKeen et al. 1994 (Edstrom 1977; Ginzberg 1979)
Ease of use	Amoako-Gyampah 2007 Igbaria and Guimaraes 1994 McGill and Klobas 2008 (Torkzadeh and Doll 1999) Wixom and Todd 2005
Data quality	Harris and Weistroffer 2009 (Zeffane et al. 1998)

Table 8 Overview of positive and negative surveys structured on a category level

Category 1	Category 2	Positive studies	Negative studies
Development Process	Development Process	(Barki and Hartwick 1994) Harris and Weistroffer 2009 (Hunton and Price 1997; Wu and Marakas 2006) (Hartwick and Barki 2001) (Hartwick and Barki 1997) (Kanungo and Bagchi 2000)	
	Human Aspects	Cavaye 1995 (Robey et al. 1989) (Gefen et al. 2008) Harris and Weistroffer 2009 (Hunton and Price 1997; Pries-Heje 2008; Wu and Marakas 2006) Ives and Olson 1984 (Alter 1978) (Kanungo and Bagchi 2000) McGill and Klobas 2008 (Amoako-Gyampah and White 1993; Barki and Hartwick 1991; Hartwick and Barki 1994)	(Barki and Hartwick 1994) Cavaye 1995 (Kim and Lee 1986; Robey and Farrow 1982)
	System Attributes	McKeen et al. 1994	
	System Success	(Bai and Cheng 2010) Cavaye 1995 (Allingham and O'Connor 1992; Baronas and Louis 1988; Baroudi et al. 1986; Jarvenpaa and Ives 1991; Kappelman and McLean 1991; Kim and Lee 1986) (Chang et al. 2010)	Cavaye 1995 (Tait and Vessey 1988) Harris and Weistroffer 2009 (Zeffane et al. 1998) Kujala 2003 (Heinbokel et al. 1996)

Table 8 (continued)

Category 1	Category 2	Positive studies	Negative studies
		(El Emam et al. 1996)	
		Harris and Weistroffer 2009 (Butler and Fitzgerald 1997; Discenza et al. 2008; Guimaraes et al. 2003; Hunton 1996; Hunton and Price 1997; Jiang et al. 2002; Kirsch and Beath 1996; Lawrence et al. 2002; Saleem 1996; Santhanam et al. 2000; Yetton et al. 2000)	
		(Kristensson et al. 2011)	
		(Kujala et al. 2005)	
		Harris and Weistroffer 2009 (Lin and Shao 2000; Lu and Wang 1997 Santhanam et al. 2000; Wu et al. 2006; Yetton et al. 2000)	
		Ives and Olson 1984 (Boland 1978; Edstrom 1977; Gallagher 1974; Guthrie 1972; King and Rodriguez 1981)	
		(Kristensson et al. 2011)	
		(Kujala et al. 2005)	
		McGill and Klobas 2008 (Amoako-Gyampah and White 1993; Barki and Hartwick 1991; Doll and Torkzadeh 1991; Doll and Torkzadeh 1988; Hawk 1993; Lawrence and Low 1993; McKeen and Guimaraes 1997; Torkzadeh and Doll 1999)	
		McKeen et al. 1994 (Edstrom 1977; Franz 1979; Ginzberg 1979; Kappelman and McLean 1991; Olson and Ives 1981)	
Human Aspects	Development Process	(Chang et al. 2010)	
		(Iivari et al. 2011)	
		Harris and Weistroffer 2009 (Wu and Marakas 2006)	
	Human Aspects	(Barki and Hartwick 1994)	
		(Chang et al. 2010)	
		Harris and Weistroffer 2009 (Lin and Shao 2000; Pries-Heje 2008)	
		Ives and Olson 1984 (Alter 1978; Igersheim 1976)	
		(Kanungo and Bagchi 2000)	
		McGill and Klobas 2008 (Amoako-Gyampah and	

Table 8 (continued)

Category 1	Category 2	Positive studies	Negative studies
		White 1993; Hartwick and Barki 1994; Jackson et al. 1997)	
	System Success	Cavaye 1995 (Doll and Torkzadeh 1989; Franz and Robey 1986; Tait and Vessey 1988)	(Amoako-Gyampah 2007)
		(Chang et al. 2010)	Harris and Weistroffer 2009 (Zeffane et al. 1998)
		(Gefen et al. 2008)	McGill and Klobas 2008 (Amoako-Gyampah and White 1993; Doll and Torkzadeh 1991)
		Harris and Weistroffer 2009 (Blili et al. 1998; Choe 1996; Foster and Franz 1999; Guimaraes et al. 2003; Guimaraes and Igbaria 1997; Hsu et al. 2008; Hunton and Price 1997; Palanisamy and Sushil 2001; Yoon et al. 1998)	
		(Igbaria and Guimaraes 1994)	
		Ives and Olson 1984 (Kaiser and Srinivasan 1980; Maish 1979; Swanson 1974)	
		(Kanungo and Bagchi 2000)	
		McGill and Klobas 2008 (Barki and Hartwick 1991; Hartwick and Barki 1994; Seddon and Kiew 1996; Torkzadeh and Lee 2003)	
		(Rouibah et al. 2008)	
		(Wixom and Todd 2005)	
Organizational Factor	Development Process	(Bai and Cheng 2010)	
		(Iivari et al. 2011)	
		Harris and Weistroffer 2009 (Lu and Wang 1997)	
	Human Aspects		McGill and Klobas 2008 (Amoako-Gyampah and White 1993)
	System Success	Cavaye 1995 (Jarvenpaa and Ives 1991)	
		Harris and Weistroffer 2009 (Guimaraes et al. 2003; Hsu et al. 2008; Lu and Wang 1997; Santhanam et al. 2000; Yetton et al. 2000)	
		McKeen et al. 1994 (Powers and Dickson 1973)	
		McGill and Klobas 2008 (Amoako-Gyampah and White 1993)	
		(Rouibah et al. 2008)	

Table 8 (continued)

Category 1	Category 2	Positive studies	Negative studies
System Attributes	Development Process	(Iivari et al. 2011) Harris and Weistroffer 2009 (Lin and Shao 2000)	
	System Attributes	McGill and Klobas 2008 (McKeen and Guimaraes 1997) McKeen et al. 1994	
	System Success	Harris and Weistroffer 2009 (Yoon et al. 1998)	(El Emam et al. 1996) Harris and Weistroffer 2009 (Palanisamy and Sushil 2001; Yetton et al. 2000)
System Success	Human Aspects	McGill and Klobas 2008 (Hartwick and Barki 1994) (Wixom and Todd 2005)	
	System Success	(Amoako-Gyampah 2007) Harris and Weistroffer 2009 (Yoon et al. 1998) (Igbaria and Guimaraes 1994) (Kristensson et al. 2011) McGill and Klobas 2008 (Rouibah et al. 2008) (Wixom and Todd 2005)	

Table 9 Aspects influenced and targeted by the methods papers

	Development Process			Human Aspects								System Attributes		Organizational Factors			System Success							
Study	User Participation	User-Developer Communication	Mode of Development	User Involvement	User's Belief about Developers	User's Attitude towards System	User's Ability in IT projects	Disagreement/Conflict	Developer's Attitude towards User	User's Intention to Use	User's Motivation	Complexity	Uncertainty	Top Management Support	Organizational or Managerial Culture	Availability of Resources	User Satisfaction	System Use	System Quality	Project in Time and Budget	Ease of Use	Data Quality	Sum	Percent
(Amoako-Gyampah and White 1997)	1	1	0	1	1	0	0	1	1	0	0	0	0	1	0	1	0	0	1	1	1	1	12	55%
(Begier 2010)	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	1	1	6	27%
(Berger 2011)	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	0	1	0	0	1	0	0	9	41%
(Bragge 2009)	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	1	0	4	18%
(Cherry and Macredie 1999)	1	1	1	0	1	1	1	1	1	0	0	0	0	1	0	0	1	1	1	0	1	0	13	59%
(Cohen et al. 2010)	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	0	0	5	23%
(Dörner et al. 2008)	1	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	1	0	0	0	5	23%
(Dean et al. 1998)	1	1	1	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	1	1	0	0	9	41%
(Eckhardt 2010)	0	1	0	0	1	0	0	0	1	0	1	0	0	0	0	0	0	1	1	1	0	0	7	32%
(Finck et al. 2004)	0	1	0	1	0	1	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	5	23%
(Fruhling et al. 2005)	1	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	1	1	0	1	0	6	27%
(Fuentes-Fernández et al. 2009)	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	2	9%
(Hansson et al. 2004)	0	1	0	1	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	12	55%
(Hansson et al. 2006)	1	1	1	1	1	1	0	0	1	0	1	0	0	0	0	0	1	1	1	0	1	0	4	18%
(Hendry 2008)	0	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	6	27%
(Hope and Amdahl 2011)	1	1	1	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	7	32%
(Huang et al. 2008)	1	0	0	0	1	0	0	0	1	1	0	0	0	0	0	0	1	0	1	0	1	0	5	23%
(Humayoun et al. 2011)	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	1	1	0	0	6	27%
(Kabbedijk et al. 2009)	1	0	0	1	0	1	0	0	0	1	0	0	0	0	0	0	1	0	1	0	0	0	7	32%
(Kamal et al. 2011)	0	0	0	1	0	0	1	0	0	0	0	1	0	1	0	0	0	0	1	1	0	1	10	45%
(Kautz 2000)	1	1	0	1	1	1	0	0	1	0	0	0	0	0	0	1	1	0	1	1	0	0	9	41%
(Kautz 2011)	1	0	1	1	0	1	0	0	1	1	0	0	0	0	0		1	0	1	1	0	0	9	41%
(Kawalek and Wood-Harper 2002)	0	1	0	1	0	1	0	1	0	1	1	1	0	0	0	0	0	0	1	1	0	0	8	36%
(Kensing et al. 1998)	1	1	0	1	1	0	1	0	1	0	0	0	0	1	0	0	1	0	0	0	0	0	3	14%
(Korkala et al. 2006)	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	7	32%
(Korkala et al. 2010)	0	1	0	1	0	0	0	0	0	0	1	1	0	0	1	1	0	0	1	0	0	0	6	27%
(Kujala 2008)	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	1	1	1	0	1	0	13	59%
(Martin et al. 2010)	1	1	1	1	1	0	1	1	1	0	1	0	0	1	0	0	1	0	1	1	0	0	8	36%
(Mambrey et al. 1998)	0	1	0	1	0	1	0	0	1	0	0	0	0	0	0	0	1	1	1	0	1	0	7	32%
(Pekkola et al. 2006)	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	6	27%
(Pérez et al. 2011)	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	1	0	12	55%
(Pries-Heje 2008)	1	1	0	1	0	1	1	1	0	0	0	1	0	0	0	1	1	1	1	1	0	0	8	36%
(Pries-Heje and Dittrich 2009)	1	1	0	1	0	1	0	0	0	0	0	1	0	1	0	0	0	0	1	1	0	0	11	50%
(Takats and Brewer 2005)	1	1	0	1	1	0	0	0	1	0	1	1	0	1	0	1	1	0	1	0	0	0	8	36%
(Teixeira et al. 2011)	1	1	0	1	0	0	0	0	0	0	0	1	0	0	0	0	1	1	1	0	1	0	7	32%
(Wagner and Piccoli 2007)	1	1	0	1	0	0	0	0	1	0	1	0	0	0	0	0	1	0	1	0	0	0	12	55%
Amount of paper per subcategory	**25**	**25**	**11**	**21**	**9**	**12**	**6**	**7**	**14**	**4**	**12**	**9**	**2**	**8**	**2**	**7**	**19**	**13**	**33**	**12**	**12**	**3**		
%per subcategory	**69 %**	**69 %**	**31 %**	**58 %**	**25 %**	**33 %**	**17 %**	**19 %**	**39 %**	**11 %**	**33 %**	**25 %**	**6 %**	**22 %**	**6 %**	**19 %**	**53 %**	**36 %**	**92 %**	**33 %**	**33 %**	**8 %**		
Amount of paper per category	**34**			**29**								**10**		**13**			**36**							
%per Category	**94%**			**81%**								**28%**		**50%**			**100%**							

Table 10 Summaries of the Method Papers

Study	Summary
(Amoako-Gyampah and White 1997)	This paper discusses the value of user involvement at all stages in software development projects in information technology. It derives most of it insight from an information system development project at a large manufacturing firm. Traditional approach to user involvement (including users on project team, setting up steering committees, using user sign-offs and provide feedback to users) are good, but not enough. In addition it is suggested to ensure: an interactive process, timely feedback on user's suggestions and input, minimizing semantic barriers between developers and users, keeping people informed about project changes, ensuring trust between project participants, ensuring effective communication, clarify roles and expectations, removing any negative perceptions.
(Begier 2010)	This paper introduces a methods how to integrate users during evolutional development of an expert system (AutoCAD system applied in civil engineering. Suggestion is to get end user feedback through a survey that can be used by developers in order to learn and improve quality of software with regards to users need (e.g., ease of use and usability).
(Berger 2011)	This paper looks into a case study within the public sector where an in house developed agile development method was used for an information system development. Even though the methods implemented all principles for user involvement within agile methods, e.g., colocation, an iterative process, joint application development (JAD) workshops, system success was not achieved. The authors explain that due to an organizational cultural mismatch that prevented a collaborative environment.
(Bragge 2009)	This paper describes two cycle of continuing action research intervention that employed collaborative engineering with e-collaboration processes. The processes should motivates end user to participate in feedback and suggest new development ideas during ongoing use of advanced web based student information system implemented in a university.
(Cohen et al. 2010)	This paper introduces a new software development methodology which they call Lean driven development. It is a suggestion for vendors of "off the shelf" software how to develop their products in line with customers and stakeholders. For the methods the authors adapted the software development life cycle model for acquisition process of information systems.
(Cherry and Macredie 1999)	This paper argues that strict requirements analysis is not very valuable for information system design. It therefore suggests participatory design approaches such as, cooperative prototypes, brainstorming, (future) workshops and organizational gaming to specify the software system design.
(Dörner et al. 2008)	This paper defines the Collaborative Software Engineering Methodology a framework for effective and efficient user involvement throughout the systems development process. This methodology includes mechanisms to support three layers of user involvement: selected user representatives, user groups (SMEs), and the entire user community. Specifically, it includes intimate involvement of individual user representatives for development of preliminary models and prototypes, groups of SMEs to refine, validate, and prioritize requirements and the broader user community during initial needs surveys and wide-scale beta testing. Participation of multiple SMEs is beneficial for both content and political reasons. Group meetings provide a useful mechanism to involve multiple SMEs. The methodology has been validated and developed during various project in public administration and defense projects.
(Dean et al. 1998)	This paper introduces a new method for self-ethnographic methods, giving infrastructure probes (snapshot tool, USB stick, digital camera, post-its, etc.) to end user and asks them to document software issues. The goal is to get a deeper understanding of the user's working context and thus help to improve the collaboration between users and developers regarding requirements elicitation.

Table 10 (continued)

Study	Summary
(Eckhardt 2010)	This paper suggests using boundary spanner, i.e. people that broke between business and IT people, in order to overcome the communication gap between IT and Business. Therefore, they interviewed two boundary spanners from which they derived different reasons for this role and created a 25 item skill set for boundary spanner.
(Finck et al. 2004)	This paper proposes a special kind of mediated feedback through a web-based groupware system called Commsy. The goal was to enable a participatory design approach, by combining the groupware's discussion forum with human facilitators. The first technique was a Facilitation-CommSy Project room was used to exchange of information about and feedback for the ongoing development and on experience in facilitating project rooms with other facilitators and the development team. The second was feedback discussions, using the discussion functionality in Project Rooms to get feedback from the working group and give an incentive as well as a platform for expressing problems and ideas about CommSy usage. The facilitator also informs users about design decisions based on requirements from discussion forum
(Fruhling et al. 2005)	This paper presents a repeatable collaborative usability testing process supported by a Group Support System (e-CUP) that was developed to involve various stakeholders in software development to increase usability and thus system success. It was evaluated in a series of workshops involving a real system called Secure Telecommunications Application Terminal Package (STATPack), which addresses critical health communication and bio-security needs. The results show that the collaborative usability testing process facilitates stakeholder involvement through stakeholder expectation management, visualization and tradeoff analysis, prioritization of usability action items, the use of advanced groupware tools, and a simple business case analysis.
(Fuentes-Fernández et al. 2009)	This paper presents an analytical tool for requirements elicitation about the human context of systems. The Activity Theory Requirements Engineering (ATRE), is built upon a well-established theory from Social Sciences, the Action Theory (AT), and standard practices of Software Engineering. ATRE abstracts and formalizes the concept of social property. A social property presents knowledge from Social Sciences that can be relevant in gaining new insights into the human context of a system. A knowledge repository stores these properties, which are organized in areas that are related to dimensions of concern in AT and aspects that refine them. This structure guides users to properties related to their current information interests. The validation within an enterprise system for a consulting firm lead to a knowledge repository that includes four areas, which contain 38 aspects with 185 properties, but can be extended with practice. Result of using the ATRE framework are requirements specifications that are more complete regarding the human context and its influence in the design and behavior of the system to be.
(Hansson et al. 2004)	This paper describes a development method from a rather small, but very successful Swedish company. The main product is a booking system developed with an agile like development methods. In order to keep the users satisfied; they have in active support were also developers answer, are holding user meeting throughout the country and have a web interface for user requirements.
(Hansson et al. 2006)	This paper describes an approach from a small Swedish software company that mixes participatory design with agile processes. It therefore uses various channels (support calls, user meetings, courses, the web site and a newsletter) to obtain feedback on further development needs from the user. The lessons learnt are also that it is important to use a Customer Relationship Management tool to keep track of user's feedback, develop a user community and use an agile-like development process.

Table 10 (continued)

Study	Summary
(Hendry 2008)	This paper addresses the question on deciding how to engage users in design and development through communication and information technology. The author develops a framework concerning user roles and discourse, which makes two claims: (1) user roles and a social structure emerge after the introduction of a software application (role differentiation); and (2) different roles demand different kinds of discourse for deciding what to do and for reflecting upon intended and unintended consequences (role discourse demands). For validation the framework is used to analyze the development of del.icio.us, a breakthrough application for social bookmarking, which uses a mailing list for user participation.
(Hope and Amdahl 2011)	This paper reviews success of two projects implementing the agile methods called Dynamic System Design Methods (DSDM) within a Norwegian software design company. DSDM aims to improve the collaboration between software designers and users, as well as develop other aspects of project management. The various activities of the DSDM-driven project include: to plan, to map end-users' needs, to describe the new system, to select architecture, to design, model, code and test, to do quality assurance and to provide project management. In one project the methods has not been understood and therefore did not really improve user participation in design in the other with the help of an external IT consultant had some success.
(Huang et al. 2008)	This paper proposes an integrated design method based on scenario and participatory design (DMSPD) for an Internet-based collaborative learning environment. The method has four phases. The first phase aims at stakeholder analysis and task analysis to acquire system requirements. Some methods, such as brainstorming, focus group, interview and questionnaire, are adopted. Scenario design is the second phase, including four subphases, viz. activity design, information design, interaction design and design specification. Design implementation is the third phase. Sketch, mock-up and prototype are used to conceive and represent design solutions in this stage. The last phase is usability testing and evaluation, consisting of usability quiz and subjective evaluation. The total design phase is an iterative process. The authors testes the method with students in the art department of a university.
(Humayoun et al. 2011)	This paper presents a framework that incorporates user-centered design (UCD) philosophy into agile software development through a three-fold integration approach: at the process life cycle level suggestions for user involvement are for elicitation (focus groups and card sort methods) and evaluation (early design within paper based prototype's, later designs with evaluation experiments run from the development environment); at the iteration level for integrating UCD concepts, roles, and activities during each agile development iteration planning; and at the development environment level for managing and automating the sets of UCD activities through automated tools support. In addition the authors present two automated tools, UEMan and TaMUlator, which provide the realization of the development-environment level integration. The paper evaluated the methods as well as their two tool in two case studies from academia.
(Kabbedijk et al. 2009)	This paper describes a customer involvement method from an ERP company that develops products for the Dutch and Belgium small and medium sized companies. They ask their customer for requirements definition. The customers can either suggest through an incident report with the support staff, they can vote electronically on requirements suggested by the company (idea feedback) or suggest own ideas for features (suggestions), which then get voted on by all customers. The last two happen at so called customer participation session, where the customers get instant feedback and align with their follow customers.
(Kamal et al. 2011)	This paper argues for the use of stakeholder theory in Technology Integration Solutions (TIS) in Local Government Authorities. It researches in four cases

Table 10 (continued)

Study	Summary
	the three areas of stakeholder analysis – namely stakeholders' identification, stakeholders' perception on the TIS adoption factors, and stakeholder involvement on the adoption life cycle phases.
(Kautz 2000)	The paper presents a successful commercial agile development project of an operations management system with an underlying ERP system, in a large German public sector organization and analyzes the case in regards to user involvement. The formalized methods includes planning techniques called planning games, user stories and story cards to specify user requirements, onsite customers to support customer-developer communication, daily, stand-up meetings of the project team to support team communication, pair programming, refactoring, collective ownership, continuous integration and testing to develop the software proper and tuning workshops to improve the development processes regularly. Furthermore the traditional XP methods was extended by some project management processes to cater for larger projects such an overall project plan, formal reporting mechanisms and a formal contract based on a requirements specification called realization concept.
(Kautz 2011)	This paper investigates how users participate in agile development. It therefore studies the methods of XP in regard to user participation. The analysis revealed that planning games, user stories and story cards, working software and acceptance test structured the participation and contributes to a successful project completion.
(Kawalek and Wood-Harper 2002)	This paper presents a study of a major, multinational program of Enterprise Systems (ES) implementation. The case study subject is a high-tech manufacturer. The investigation looks into the fact that the implementation methods espoused user participation even though the outcome of the project was already known (regardless of user input, the ES would be deployed as a standard system). The paper reports that user participation was deployed to serve the interests of the project manager in reporting local circumstances as the implementation project moved across different sites. The users reported positively that they were involved at all and thus the enhance communication prevented conflicts. The framework for this inquiry was Multiview2, the latest generation of the Multiview information systems methods. The structure of Multiview2 was used as a diagnostic device in order to inquire into the characteristics of the ES methods used at the case study sites.
(Kensing et al. 1998)	This paper describes a participatory design method called MUST which has been developed throughout 10 different projects. It is based on five principles and suggests the five main activities (project establishment, strategic analysis, in-depth analysis of selected work domains, developing visions of the overall change, anchoring the visions), and some techniques for each activity, e.g., meetings, document analysis, future workshops, are proposed.
(Korkala et al. 2006)	This paper compares the use of different communication media within four case studies using Mobile-D development approach inspired by several agile methods. They found out aligned with the Media Richness Theory that reliance on lean media (e.g., e-mail and telephone) led to a much higher defect rate. Thus they suggest using rich media (e.g., face-to-face meetings or videoconferencing) especially for requirements analysis with light-weighted prototype, mid-iteration communication and iteration planning. For negations, telephone can be used if the user is not on site. Rich media communication ensures a low defect rate and therefore higher quality software.
(Korkala et al. 2010)	This paper looks into distributed development that combines traditional and agile methodologies with regard to customer communication. It provides

Table 10 (continued)

Study	Summary
	practical guidelines for companies in distributed agile environments, such as to define a person to play the role of the customer, if face-to-face, synchronous communication is infeasible, use an e-mail listserv to increase the chance of a response and encourage prompt, useful, and conclusive responses to e-mails and use a globally-available project management tools to record and monitor the project status on a daily basis. The key finding from a qualitative case study was that it might be difficult for an agile organization to get relevant information from a traditional type of customer organization, even though the customer communication was indicated to be active and utilized via multiple different communication media.
(Kujala 2008)	This paper suggests using more field studies to improve early user participation in product development. Field studies appear as a promising approach, but the analysis of the gathered user needs has been shown to be demanding. This study presents, on the basis of seven case studies, an early user-involvement process (Identify stakeholders and user groups—Visit users and explore their needs—Describe the current situation—Analyze and prioritize the problems and possibilities—Redesign the current situation—Define user requirements). The process is evaluated in two industrial cases for software managing infrastructures with interviews and a questionnaire. The results show that the process supports effective early user involvement; the resulted requirements were evaluated as being more successful and their quality as better than average in a company.
(Martin et al. 2010)	This paper extensively studies role of customer in agile XP projects. Based in eleven case studies from various industries, the author derives roles for collaboration (Geek Interpreter, Political Advisor and Technical Liaison), roles based on skills to support onsite customers (Acceptance Tester, UI Designer and Technical Writer) and roles for direction setting (Negotiator, Diplomat, Super-Secretary and Coach). Furthermore the author identifies practices which primarily support Real Customer Involvement by preparing the business representatives for their role (Customer Boot Camp), and providing opportunities for the business representatives to contribute of what to build (Big Picture Up-Front, Road show and Re-calibration). In addition it is important for the programmers to develop empathy for the customer team (Customer's Apprentice) and the end-user (Programmer On-Site).
(Mambrey et al. 1998)	This paper describes participatory design activities within the PoliTeam project. In this introduction of a groupware system into the German government, the project used user advocacy (User advocates augment interaction between users and designers) and osmosis (multi-level information that a designer receives by visiting the workplace and having contact with users) in an evolutional cycling process.
(Pekkola et al. 2006)	This paper suggests an iterative information system development process that uses participatory design (PD) elements. Especially the use of prototypes in various form lead to better requirements and thus more user orientation. The process has 8 steps, which are: 1. Decision to start ISD process (gaining commitment to PD by all project participants) 2. Outlining preliminary user requirements (sharpen user requirements with scenarios, paper prototypes) 3. Analyzing and designing prototype (testing system concept, components and UI) 4. Implementing full-feature prototype 5. Introducing prototype into the organization 6. Gather suggestions for improvements 7. Analyzing and verifying the requirements (select features for next version) 8. Finish system or prototype

Table 10 (continued)

Study	Summary
(Pérez et al. 2011)	This paper presents a method that involves end users within Model Driven Development (MDD) approaches to ensure that the application fits the end users' expectation. The method follows good practices and techniques in End User Development, and combines it with modeling techniques. The authors applied the methods within an existing MDD approach named PervML for allowing end users to participate in the description of their smart home. End users can now participate in the software development by means of an appropriate Domain Specific Visual Language and specific tool support for them.
(Pries-Heje 2008)	This paper mainly reports on a large ERP project at the Danish headquarter of an international engineering company that consciously used user participation during the whole implementation process. Even though the user of the consulting company were heavily involved (e.g., responsible for requirements definition) within the process the dynamic switch which lead to "Pseudo-Participation" (a situation where users are asked to participate, but not given the possibility to influence the design). Nevertheless after "go-live" in the follow up phase quality and usability issue could be resolved. The author suggests having a thorough requirements specification, as a basis for the contract, allowing necessary customizations to the system, choosing an implementation partner recognizing the uniqueness of the organizational way of operation and having users and consultants work as one team. Furthermore, the companies should be aware who influences the design (standard system or organization) and should think about tools and techniques how to support the users' gain of knowledge about technical and socio-technical options.
(Pries-Heje and Dittrich 2009)	This paper describes an in-depth case study of an ERP project and its challenges in user participation and involvement. It suggests PD approaches that could have prevented some of misunderstandings that occurred during the project. These four approaches are: mediating cooperative design, shared representations, prototypes and iterative design, and have an ERP competence center to mediate between users and external IT experts.
(Takats and Brewer 2005)	This paper presents four pattern that help to improve customer/user developer relationship by extending agile methods. As customers rarely can work all the time with developers they include a workshop series into the iterative agile approach. The workshops were highly facilitated and focused on establishing a vision and high-level requirements (deliverables: Program vision, Logical Architecture, Executive briefing, Operational Concept Diagram and a Capability timeline). They used pattern from the development company call group solve, be visual and forces rank.
(Teixeira et al. 2011)	This paper describes the development of an interactive health information system called hemo@care with the help of user-centered design and participatory design practices. It suggests the three phases exploratory (analysis of documentation, direct observation and focus group), design (object-oriented system analysis (OOSA), hierarchical task analysis and prototyping), and coding (eXtremProgramming)
(Wagner and Piccoli 2007)	This paper argues based on a case study of an ERP system at a university which strictly followed the tenets of participatory design, that end users are only interested in real participation, when the system affects their everyday life, i.e., after go-live. Therefore, they suggest to think differently about how to involve users (i.e., starting point the present moment by asking users about their day-to-day work activities to elicit users' stories), broaden the skill set of system analysts (i.e., interpreting user's narratives) and enact a modified system development life cycle (i.e., recognize that implementation extends beyond "go-live").

References

Abelein U, Paech B (2012) A proposal for enhancing user-developer communication in large IT projects. Proceedings of the 5th international workshop on cooperative and human aspects of software engineering (CHASE 2012) at the ICSE 2012 Zurich, June 2nd. doi:978-1-4673-1824-2

Alleman G (2002) Agile project management methods for ERP : how to apply agile processes to complex COTS projects and live to tell about it. In: Wells D, Williams L (eds) Extreme programming and agile methods: XP/Agile Universe. Springer Verlag, pp 70–88

Allingham P, O'Connor M (1992) MIS success: why does it vary among users&quest. J Inf Technol 7:160–168

Al-Rawas A, Easterbrook S (1996) Communication problems in requirements engineering: a field study. Proceedings of the first Westminster conference on professional awareness in software engineering. Citeseer, pp 1–12

Alter S (1978) Development patterns for decision support systems. MIS Q Q 2:33–42

Amoako-Gyampah K (2007) Perceived usefulness, user involvement and behavioral intention: an empirical study of ERP implementation. Comput Hum Behav 23:1232–1248. doi:10.1016/j.chb.2004.12.002

Amoako-Gyampah K, White K (1993) User involvement and user satisfaction: an exploratory contingency model. Inf Manag 25:25–33

Amoako-Gyampah K, White K (1997) When is user involvement not user involvement? Inf Strateg: Exec J 13:40–45

Bai H, Cheng J (2010) The impact of organizational culture on ERP assimilation: the mediating role of user participation. 2010 2nd international workshop on database technology and applications. IEEE, pp 1–5

Bano M, Zowghi D (2013) User involvement in software development and system success: a systematic literature review. EASE'13 Proceedings of the 17th international conference on evaluation and assessment in software engineering 125–130

Barki H, Hartwick J (1994) User participation, conflict, and conflict resolution: the mediating roles of influence. Inf Syst Res 5:422–438. doi:10.1287/isre.5.4.422

Barki H, Hartwick J (1991) User participation and user involvement in information system development. Proceedings of the 24th annual Hawaii International conference on system sciences. pp 487–492

Baronas A-MK, Louis MR (1988) Restoring a sense of control during implementation: how user involvement leads to system acceptance. MIS Q 12:111–126. doi:10.2307/248811

Baroudi JJ, Olson MH, Ives B (1986) An empirical study of the impact of user involvement on system usage and information satisfaction. Commun ACM 29:232–238

Beck K, Beedle M, Van Bennekum A, et al. (2001) Manifesto for agile software development. The agile alliance

Begier B (2010) Evolutionally improved quality of intelligent systems following their users' point of view. Advances in Intelligent Information and Database Systems Studies in Computational Intelligence. 283:191–203

Berger H (2011) Reframing humans in information systems development. Inf Syst 201:263–281. doi:10.1007/978-1-84996-347-3

Bjarnason E, Wnuk K, Regnell B (2011) Requirements are slipping through the gaps—A case study on causes & effects of communication gaps in large-scale software development. 2011 IEEE 19th international requirements engineering conference. IEEE, pp 37–46

Blili S, Raymond L, Rivard S (1998) Impact of task uncertainty, end-user involvement, and competence on the success of end-user computing. Inf Manag 33:137–153. doi:10.1016/S0378-7206(97)00043-8

Boland R (1978) The process and product of system design. Manag Sci 24:887–898

Bragge J (2009) Engineering E-collaboration processes to obtain innovative end-user feedback on advanced web- based information systems. J Assoc Inf Syst 10:196–220

Butler T, Fitzgerald B (1997) A case study of user participation in the information systems development process. Proceesdings of the 18th international conference on information systems. pp 411–426

Cavaye A (1995) User participation in system development revisited. Inf Manag 28:311–323

Chang K, Shin T, Klein G et al (2010) User commitment and collaboration: motivational antecedents and project performance. Inf Softw Technol 52:672–679. doi:10.1016/j.infsof.2010.02.003

Cherry C, Macredie R (1999) The importance of context in information system design: an assessment of participatory design. Requir Eng 4:103–114. doi:10.1007/s007660050017

Choe J (1996) The relationships among performance of accounting information systems, influence factors, and evolution level of information systems. J Manag Inf Syst 12:215–239

Cohen S, Dori D, De HU (2010) A software system development life cycle model for improved stakeholders' communication and collaboration. Int J Comput Commun Control 5:20–41

Curtis B, Krasner H, Iscoe N (1988) A field study of the software design process for large systems. Commun ACM 31:1268–1287

Davis F, Bagozzi R, Warshaw P (1989) User acceptance of computer technology: a comparison of two theoretical models. Manag Sci 35:982–1002. doi:10.1287/mnsc.35.8.982
Dean D, Lee J, Pendergast M et al (1998) Enabling the effective involvement of multiple users : methods and tools for collaborative software engineering. J Manag Inf Syst 14:179–222
De Brabander B, Thiers G (1984) Successful information system development in relation to situational factors which affect effective communication between MIS-users and EDP-specialists. Manag Sci 30:137–155
Discenza R, Tesch D, Klein G, Jiang JJ (2008) User involvement to enhance expertise in system development. Int J Internet Enterp Manag 5:373–389. doi:10.1504/IJIEM.2008.020107
Doll W, Torkzadeh G (1991) A congruence construct of user involvement. Decis Sci 22:443–453
Doll W, Torkzadeh G (1988) The measurement of end-user computing satisfaction. MIS Q 12:259–274
Doll WJ, Deng X (2001) The collaborative use of information technology: end-user participation and system success. Inf Resour Manag J 14:6–16. doi:10.4018/irmj.2001040101
Doll WJ, Torkzadeh G (1989) A discrepancy model of end-user computing involvement. Manag Sci 35:1151–1171
Dörner C, Heß J, Pipek V (2008) Fostering user-developer collaboration with infrastructure probes. Proceedings of the 2008 international workshop on cooperative and human aspects of software engineering—CHASE'08 48–44. doi: 10.1145/1370114.1370126
Eckhardt A (2010) Lost in Translation ?!—The need for a boundary spanner between Business and IT. SIGMIS-CPR'10, May 20–22, 2010, Vancouver, BC, Canada. pp 75–82
Edstrom A (1977) User influence and the success of MIS projects: a contingency approach. Hum Res 30:589–607. doi:10.1177/001872677703000702
El Emam K, Quintin S, Madhavji N (1996) User participation in the requirements engineering process: an empirical study. Requir Eng 1(1):4–26. doi:10.1007/BF01235763
Finck M, Gumm D, Pape B (2004) Using groupware for mediated feedback. Proceedings of the eighth conference Biennial participatory design conference 2004: Artful integration: interwearing media, materials and practices—volume 2, July 27—July 7, 2004, Toronto, Canada. doi: 0-9667818-3-X
Foster ST, Franz CR (1999) User involvement during information systems development: a comparison of analyst and user perceptions of system acceptance. J Eng Technol Manag 16:329–348. doi:10.1016/S0923-4748(99)00014-4
Franz C (1979) Contingency factors affecting the user involvement role in the design of successful information systems
Franz CR, Robey D (1986) Organizational context, user involvement, and the usefulness of information systems. Decis Sci 17:329–356. doi:10.1111/j.1540-5915.1986.tb00230.x
Fruhling A, de Vreede GJ, Vreede G-J (2005) Collaborative usability testing to facilitate stakeholder involvement. Value Based Softw Eng 201–223
Fuentes-Fernández R, Gómez-Sanz J, Pavón J (2009) Understanding the human context in requirements elicitation. Requir Eng 15:267–283. doi:10.1007/s00766-009-0087-7
Gallagher C (1974) Perceptions of the value of a management information system. Acad Manag J 17:46–55
Gefen D, Ragowsky A, Ridings C (2008) Leadership and justice: increasing non participating users' assessment of an IT through passive participation. Inf Manag 45:507–512. doi:10.1016/j.im.2008.08.001
Ginzberg MJ (1979) A study of the implementation process. TIMS studies in the management sciences 85–102
Guimaraes T, Igbaria M (1997) Client/server system success: exploring the human side. Decis Sci 28:851–876
Guimaraes T, Staples D, McKeen J (2003) Empirically testing some main user-related factors for systems development quality. Qual Manag J 10:39–54
Guthrie A (1972) A survey of Canadian middle managers' attitudes toward management information systems
Hansson C, Dittrich Y, Randall D (2004) Agile processes enhancing user participation for small providers of off-the-shelf software. World 175–183
Hansson C, Dittrich Y, Randall D (2006) How to include users in the development of off-the-shelf software: a case for complementing participatory design with agile development. Proceedings of the 39th annual Hawaii international conference on system sciences (HICSS'06). IEEE, p 175c–175c
Harris M, Weistroffer H (2009) A New look at the relationship between user involvement in systems development and system success development and system success. Commun Assoc Inf Syst 24:739–756
Hartwick J, Barki H (2001) Communication as a dimension of user participation, Professional Communication. IEEE Transactions 44:21–36. doi:10.1109/47.911130
Hartwick J, Barki H (1994) Explaining the role of user participation in information system use. Manag Sci 40:440–465. doi:10.1287/mnsc.40.4.440
Hartwick J, Barki H (1997) Delineating the dimensions of user participation: a replication and extension. Review literature and arts of the Americas
Hawk S (1993) The effects of user involvement: some personality determinants. Int J Man–Mach Stud 38:839–855

Hazzan O, Tomayko J (2004) Human aspects of software engineering: the case of extreme programming. Proceedings of the 5th international conference, XP 2004, Garmisch-Partenkirchen, Germany, June 6–10. pp 303–311

Heinbokel T, Sonnentag S, Frese M et al (1996) Don't underestimate the problems of user centredness in software development projects - there are many! Behav Inform Technol 15:226–236

Hendry D (2008) Public participation in proprietary software development through user roles and discourse. Int J Hum Comp Stud 66:545–557. doi:10.1016/j.ijhcs.2007.12.002

Hirschheim R (1985) User experience with and assessment of participative systems design. MIS Q 9:295–303

Hope K, Amdahl E (2011) Configuring designers? Using one agile project management methodology to achieve user participation. N Technol Work Employ 26:54–67

Hsu LL, Lai RSQ, Te Weng Y (2008) Understanding the critical factors effect user satisfaction and impact of ERP through innovation of diffusion theory. Int J Technol Manag 43:30–47. doi:10.1504/IJTM.2008.019405

Huang Y, Lu R, Sun S (2008) Designing a cooperative learning system: a scenario and participatory design based approach. Int Symp Computat Intell Des 2008:334–337. doi:10.1109/ISCID.2008.203

Humayoun S, Dubinsky Y, Catarci T (2011) A three-fold integration framework to incorporate user – centered design into agile software development. Hum Centered Des HCII 6776:55–64. doi:10.1007/978-3-642-21753-1_7

Hunton J (1996) Involving information system users in defining system requirements: the influence of procedural justice perceptions on user attitudes and performance. Decis Sci 27:647–671. doi:10.1111/j.1540-5915.1996.tb00868.x

Hunton J, Beeler J (1997) Effects of user participation in systems development: a longitudinal field experiment. MIS Q 21:359–388. doi:10.2307/249719

Hunton JE, Price KH (1997) Effects task of the user participation process outcomes and meaningfulness on Key information system. Manag Sci 43:797–812

Hwang M, Thorn R (1999) The effect of user engagement on system success: a meta-analytical integration of research findings. Inf Manag 35:229–236. doi:10.1016/S0378-7206(98)00092-5

Igbaria M, Guimaraes T (1994) Empirically testing the outcomes of user involvement in DSS development. Omega 22:157–172

Igersheim RH (1976) Managerial response to an information system. Conference proceedings of AFIPS. ACM Press, New York, New York, USA, pp 877–882

Iivari J, Igbaria M, Taylor P (2011) Determinants of user participation: a Finnish survey. Behav Inform Technol 16:37–41. doi:10.1080/014492997119950

Iivari J, Isomäki H, Pekkola S (2010) The user - the great unknown of systems development: reasons, forms, challenges, experiences and intellectual contributions of user involvement. Inf Syst J 20:109–117. doi:10.1111/j.1365-2575.2009.00336.x

Iivari N (2004) Enculturation of user involvement in software development organizations - an interpretive case study in the product development context. Proceedings of the third Nordic conference on human-computer interaction—NordiCHI'04 287–296. doi: 10.1145/1028014.1028059

Ives B, Olson M (1984) User involvement and MIS success: a review of research. Manag Sci 30:586–603

Jackson CM, Chow S, Leitch RA (1997) Toward an understanding of the behavioral intention to use an information system. Decis Sci 28:357–389. doi:10.1111/j.1540-5915.1997.tb01315.x

Jalali S, Wohlin C (2012) systematic literature studies: database searches vs. Backward Snowballing. Proceedings of the International conference on empirical software engineering and measurement, ESEM'12. pp 29–38

Jarvenpaa S, Ives B (1991) Executive involvement and participation in the management of information technology. MIS Q 15:205–224

Jiang JJ, Chen E, Klein G (2002) The importance of building a foundation for user involvement in information system projects. Proj Manag J 33:20–26

Jorgensen M, Dyba T, Kitchenham B (2005) Teaching evidence-based software engineering to university students. 11th IEEE International software metrics symposium. IEEE, p 24

Kabbedijk J, Brinkkemper S, Jansen S, van der Veldt B (2009) Customer involvement in requirements management: lessons from mass market software development. In: Brinkkemper S, Jansen S, Veldt B van der (eds) IEEE international conference on requirements engineering. IEEE, pp 281–286

Kaiser KM, Srinivasan A (1980) The relationship of user attitudes toward design criteria and information systems success. National AIDS conference proceedings. pp 201–203

Kamal M, Weerakkody V, Irani Z (2011) Analyzing the role of stakeholders in the adoption of technology integration solutions in UK local government: an exploratory study. Gov Inf Q 28:200–210. doi:10.1016/j.giq.2010.08.003

Kanungo S, Bagchi S (2000) Understanding user participation and involvement in ERP Use. J Manag Res 1:47–64

Kappelman L, McLean E (1991) The Resepective roles of user participation and user involevemnt in information systenm implementation success. Proceedings of the twelfth international conference on Information systems. pp 339–349

Kautz K (2000) Customer and User Involvement in Agile Software Development. Analysis 168–173. doi: 10.1007/978-3-642-01853-4_22

Kautz K (2011) Investigating the design process: participatory design in agile software development. Inf Technol People 24:217–235. doi:10.1108/09593841111158356

Kawalek P, Wood-Harper T (2002) The finding of thorns : user participation in enterprise system implementation. Data Base Adv Inf Syst 33:13–22

Kensing F, Simonsen J, Bodker K (1998) MUST: A method for participatory design. Hum Comput Interact 13:129–140

Kim E, Lee J (1986) An exploratory contingency model of user participation and MIS use. Inf Manag 11:87–97

King W, Rodriguez J (1981) Participative design of strategic decision support systems: an empirical assessment. Manag Sci 27:717–726

Kirsch L, Beath C (1996) The enactments and consequences of token, shared, and compliant participation in information systems development. Acc Manag Inf Technol 6:221–253

Kitchenham B, Charters S (2007) Guidelines for performing systematic literature reviews in software engineering. doi: 10.1.1.117.471

Korkala M, Abrahamsson P, Kyllönen P (2006) A case study on the impact of customer communication on defects in agile software development . In: Abrahamsson P, Kyllonen P (eds) AGILE 2006 (AGILE'06). IEEE, pp 76–88

Korkala M, Pikkarainen M, Conboy K (2010) Combining agile and traditional: Customer communication in distributed environment. In: Šmite D, Moe NB, Ågerfalk PJ (eds) Agility across time and space. Springer Berlin Heidelberg, Berlin, Heidelberg, pp 201–216

Kristensson P, Gustafsson A, Witell L (2011) Collaboration with customers - understanding the effect of customer-company interaction in new product development. 2011 44th Hawaii international conference on system sciences. IEEE, pp 1–9

Kujala S (2008) Effective user involvement in product development by improving the analysis of user needs. Behav Inform Technol 27:457–473. doi:10.1080/01449290601111051

Kujala S (2003) User involvement: a review of the benefits and challenges. Behav Inform Technol. doi:10.1080/01449290210000555530

Kujala S, Kauppinen M, Lehtola L, Kojo T (2005) The role of user involvement in requirements quality and project success. 13th IEEE International Conference on Requirements Engineering (RE'05) 75–84. doi: 10.1109/RE.2005.72

Kujala S, Taylor P, Kujala S (2008) Effective user involvement in product development by improving the analysis of user needs. Behav Inform Technol 27:457–473. doi:10.1080/01449290601111051

Lawrence M, Goodwin P, Fildes R (2002) Influence of user participation on DSS use and decision accuracy. Omega 30:381–392

Lawrence M, Low G (1993) Exploring individual user satisfaction within user-led development. Manag Inf Syst Q 17:195–208

Lin WT, Shao BBM (2000) The relationship between user participation and system success: a simultaneous contingency approach. Inf Manag 37:283–295. doi:10.1016/S0378-7206(99)00055-5

Lu H-P, Wang J-Y (1997) The relationships between management styles, user participation, and system success over MIS growth stages. Inf Manag 32:203–213. doi:10.1016/S0378-7206(97)00021-9

Lucas (1975) Why information systems fail

Lynch T, Gregor S (2004) User participation in decision support systems development: influencing system outcomes. Eur J Inf Syst 13:286–301

Maalej W, Happel H-JHJ, Rashid A (2009) When users become collaborators: towards continuous and context-aware user input. Proceeding of the 24th ACM SIGPLAN conference companion on object oriented programming systems languages and applications. ACM, pp 981–990

Maalej W, Pagano D (2011) On the Socialness of Software. 2011 IEEE ninth international conference on dependable, autonomic and secure computing. IEEE, pp 864–871

Maish A (1979) A User's behavior toward his MIS. MIS Q 3:39–52

Majid R, Noor N, Adilah W, et al. (2010) A survey on user involvement in software Development Life Cycle from practitioner's perspectives. 5th international conference on computer sciences and convergence information technology. IEEE, pp 240–243

Mambrey P, Mark G, Pankoke-Babatz U (1998) User advocacy in participatory design : designers' experiences with a new communication channel. Computer Supported Cooperative Work (CSCW) 7:291–313. doi: 1008687122083

Martin A, Biddle R, Noble J (2010) An ideal customer: a grounded theory of requirements elicitation, communication and acceptance on agile projects. Agile software development: current research and future directions. Springer, Berlin, pp 111–141

McGill T, Klobas J (2008) User developed application success: sources and effects of involvement. Behav Inform Technol 27:407–422. doi:10.1080/01449290601110715

McKeen J, Guimaraes T (1997) Successful strategies for user participation in systems development. J Manag Inf Syst 14:133–150

McKeen J, Guimaraes T, Wetherbe J (1994) The relationship between user participation and user satisfaction: an investigation of four contingency factors. MIS Q 18:427–451. doi:10.2307/249523

Olson M, Ives B (1981) User involvement in system design: an empirical test of alternative approaches. Inf Manag 4:183–196

Palanisamy R, Sushil (2001) User involevemnt in information systems planning leads to strategic success: an empirical study. J Serv Res 1:125–150

Pekkola S, Kaarilahti N, Pohjola P (2006) Towards formalised end-user participation in information systems development process : Bridging the gap between participatory design and ISD Methodologies. Information Systems 21–30.

Pérez F, Valderas P, Fons J (2011) Towards the involvement of End-users within model-driven development. International conference on End-user development. Springer, Berlin, pp 258–263

Powers R, Dickson G (1973) MIS project management: myths, opinions, and reality. Calif Manag Rev 15:147–156

Pries-Heje L (2008) Time, attitude, and user participation: how prior events determine user attitudes in ERP implementation. Int J Enterp Inf Syst 4:48–65. doi:10.4018/978-1-60566-968-7.ch006

Pries-Heje L, Dittrich Y (2009) ERP implementation as design. Scand J Inf Syst 21:27–58

Robey D, Farrow D (1982) User involvement in information system development: a conflict model and empirical test. Manag Sci 28:73–85. doi:10.1287/mnsc.28.1.73

Robey D, Farrow DL, Franz CR (1989) Group process and conflict in system development. Manag Sci 35:1172–1191. doi:10.1287/mnsc.35.10.1172

Rouibah K, Hamdy HI, Al-Enezi M (2008) Effect of management support, training, and user involvement on system usage and satisfaction in Kuwait. Industrial Management & Data Systems 109:338–356. doi:10.1108/02635570910939371

Saleem N (1996) An empirical test ofthe contingency approach to user participation in information systems development. J Manag Inf Syst 13:145–166

Santhanam R, Guimaraes T, George JF (2000) An empirical investigation of ODSS impact on individuals and organizations. Decis Support Syst 30:51–72. doi:10.1016/S0167-9236(00)00089-0

Seddon P, Kiew M (1996) A partial test and development of DeLone and McLean's model of IS success. Aust J Inf Syst 4:90–109

Singh S, Kotzé P (2003) An overview of systems design and development methodologies with regard to the involvement of users and other stakeholders. Proceedings of SAICSIT 2003. pp 37–47

Sommerville I (2007) Software engineering. Addison-Wesley

Subramanyam R, Weisstein FL, Krishnan MS (2010) User participation in software development projects. Commun ACM 53:137. doi:10.1145/1666420.1666455

Sutcliffe A, Economou A, Markis P (1999) Tracing requirements errors to problems in the requirements engineering process. Requir Eng 4:134–151. doi:10.1007/s007660050024

Swanson E (1974) Management information systems: appreciation and involvement. Manag Sci 21:178–188

Tait P, Vessey I (1988) The effect of user involvement on system success: a contingency approachs. Manag Inf Syst 12:91–108

Takats A, Brewer N (2005) Improving Communication between Customers and Developers. Agile Development Conference/Australasian Database Conference 0:243–252. doi: 10.1109/ADC.2005.30

Teixeira L, Saavedra V, Ferreira C, Santos B (2011) Using participatory design in a health information system. Conference proceedings : annual international conference of the IEEE engineering in medicine and biology society 2011:5339–42. doi: 10.1109/IEMBS.2011.6091321

Torkzadeh G, Doll W (1999) The development of a tool for measuring the perceived impact of information technology on work. Omega Int J Manag Sci 27:327–339. doi:10.1016/S0305-0483(98)00049-8

Torkzadeh G, Lee J (2003) Measures of perceived end-user computing skills. Inf Manag 40:607–615. doi:10.1016/S0378-7206(02)00090-3

Wagner E, Piccoli G (2007) Moving beyond user participation to achieve successful IS design. Commun ACM 50:51–55. doi:10.1145/1323688.1323694

Wixom B, Todd P (2005) A theoretical integration of user satisfaction and technology acceptance. Inf Syst Res 16:85–102. doi:10.1287/isre.1050.0042

Wu J-T, Marakas G (2006) The impact of operational user participation on perceived system implementation success: an empirical investigation. J Comput Inf Syst 47:127–140

Yetton P, Martin A, Sharma R, Johnston K (2000) A model of information systems development project performance. Inf Syst J 10:263–289

Yoon Y, Guimaraes T, Clevenson A (1998) Exploring expert system success factors for business process reengineering. J Eng Technol Manag 15:179–199. doi:10.1016/S0923-4748(98)00011-3

Zeffane R, Cheek B, Meredith P (1998) Does user involvement during information systems development improve data quality? Hum Syst Manag 17:115–121

Dipl. – Inform. Ulrike Abelein is a PhD student in the Department of Software Engineering at the University of Heidelberg since September 2011. Her research focus is on communication between enduser and developer in large-scale IT projects. She holds a Master of Science degree in Computer Science from the Karlsruhe Institute of Technology (formally known as University of Karlsruhe).

Prof. Dr. Barbara Paech is a professor for Software Engineering at the University of Heidelberg. Her teaching and research focuses on methods and processes to ensure quality of software with adequate effort. She is particularly active in the area of requirements and rational engineering. Based on her experiences as department head at the Fraunhofer Institute Experimental Software Engineering her research is often empirical and in close cooperation with industry.