Editor: **Tore Dybå**
SINTEF
tore.dyba@sintef.no

Editor: **Helen Sharp**
The Open University, London
h.c.sharp@open.ac.uk

Does Involving Users in Software Development Really Influence System Success?

Ulrike Abelein, Helen Sharp, and Barbara Paech

USERS OF SOFTWARE systems are considered to be an important source of information for software development; they might, for example, provide requirements, test finished code, and evaluate prototypes. Users are familiar with both the work and the context that the software system should support; thus it's critical to involve them in the process.

Researchers have studied how best to involve users in software development for a long time, primarily in the area of information systems (IS) and human-computer interaction (HCI).[1] In 1989, Henri Barki and Jon Hartwick defined two separate but related terms.[2] User *participation* happens when the end user takes an active part in the development or design process together with the developer—that is, user participation refers to the "behaviors and activities users perform in the system development process." User *involvement*, on the other hand, is more on the mental level, referring to "the psychological state of the individual, defined as the importance and personal relevance of a system to a user." Here, we consider the effects of both user participation and involvement, which we abbreviate to UPI so that we can refer to both terms together. Existing research describes several benefits of UPI, such as improved quality due to more precise requirements, the prevention of unneeded and expensive features, and an increase in user satisfaction, which leads to higher system use.[3,4] But even though some researchers consider it to be essential to system success, other studies have found contradicting results. Furthermore, it's not a common practice in today's IT projects to involve users to a large extent—in particular, large-scale projects that follow traditional software development methods involve users only to a limited degree.

To clarify UPI's effects on system success and to get a deeper understanding of the differences between user participation and user involvement, we reviewed the existing UPI literature in software development and conducted a systematic mapping study (and the full review can be accessed elsewhere[5]). The review revealed 16 statistical surveys and six metastudies that empirically research various UPI effects on system success. Including all reference studies from the metastudies,

we analyzed correlation data from 86 studies.

How to Involve Users

Researchers have devised many approaches for increasing UPI in software development, and some software development methods provide multiple opportunities and mechanisms for users to be involved. The most common approaches are participatory design, user-centered design, ethnography, and contextual design.[6] They differ primarily in user activity and whether users actually take part in decision making.

Participatory design, for example, originated in Scandinavia and emphasizes democracy and skill enhancement. Users are part of the decision-making process through workshops or prototype evaluation. User-centered design comes from the HCI field and puts the user—not technical needs—into the center of design. Based on John Gould and Clayton Lewis, the three principles of user-centered design are an early focus on users and tasks, empirical measurement based on simulation or prototypes, and iterative design.[7] Ethnography uses observations or video analysis to understand work practices and context, so users are involved but don't actively participate in decision making. Contextual design emphasizes the context of use for the system and is underpinned by contextual inquiry, which combines observation, discussion, and reconstruction of past events.

In addition to these UPI-specific approaches, software development methods provide various opportunities to include users. 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. Therefore, users are typically involved only in the system's requirements definition and validation stages. Rapid application development consists of the requirements planning, user design, construction, and cutover phases. Short development cycles ensure a close match of the system to business needs, with UPI achieved throughout design and development via prototype evaluation.[8] This approach has its disadvantages from a technical perspective, such as the lack of code reuse or program consistency, which prevents it from being used in large-scale projects. Agile or lightweight development approaches have evolutionary and incremental life cycles and use iterative development and intensive stakeholder involvement. The dynamic systems development method (DSDM), scrum, and extreme programming (XP) embrace unstable business needs and use flexible development and short implementation cycles to mitigate risks, so continuous feedback to and from the user is central.

In fact, these software development methods and UPI approaches aren't orthogonal. For example, it's possible to use participatory design within traditional development approaches or combine it with agile methods, although it might not be straightforward.

Empirical Evidence for UPI's Influence on Success

In 2012, we undertook a systematic mapping study on the influence of user participation and involvement on system success to systematically identify and meta-analyze the existing empirical evidence. As mentioned earlier, we wanted to understand which aspects have been studied and the positive and negative correlations reported. Our review identified 3,689 hits from which we identified 22 papers (16 surveys and six metastudies) that reported correlations between aspects of UPI and system success. For our meta-analysis, we also included the secondary studies of the metastudies, thus

TABLE 1

Definitions of categories and subcategories.

Category/subcategory	Definition for study	Number of studies
Development process	Different activities of project participants (end users, developers) that contribute to system development	
User participation	Behavior and activities users perform in system development process, such as being the project team leader, having responsibility for overall system success, and being responsible for selecting hardware or software, estimating costs, and requesting funds	52
User-developer communication	Communication, evaluation, and approval activities that take place between users and IS staff, as well as the frequency, content, and direction of that communication	13
Mode of development	Depending on which roles are responsible for development, the process can vary: the system can be developed by developers, end users directly, or in a cooperative way between these groups	4
Human aspects	Attitude or beliefs of project participant	
User involvement	Psychological state of the individual; the importance and personal relevance of a system to a user (also refers to the degree of users' perception on their sense of ownership toward the system)	32
User's attitude toward system	Affective or evaluative judgement of the user toward the system; the extent to which the user feels the system is good or bad	11
User's beliefs about developers	Attitude and beliefs of users regarding the behavior of developers, such as whether they take users seriously or think the decision process is fair	6
User's ability in IT projects	Ability that enables users to participate as a member of the system development team and accomplish goals of project	5
Disagreement/conflict	Divergence of opinions and goals that can lead to conflicts, as well as their resolution	4
User's intention to use	A function of attitudes toward a behavior and subjective norms that has been found to predict actual behavior	3
Developer's attitude toward user	Attitude of system developers toward users; treating users with dignity, for example	2
User's motivation	Motivation is a rationally calculative perspective that an individual's involvement in an activity arises from his or her desire to obtain rewards, including the instrumentality of creating opportunity and improving work conditions	2

using significant correlation data from 86 unique studies.

Classifying User Participation and Involvement

To understand the different aspects of UPI described in these studies, we developed a classification of them; see Table 1 for a list. This was necessary because the identified studies used more than 200 different aspects. We wanted to follow Barki and Hartwick's separation of the terms participation and involvement, so we extended their ideas to generate two main categories of UPI aspects: development process and human aspects. The development process category includes all aspects of UPI that concern active participation in the software development process and the team's roles and responsibilities, as well as communication with the people involved in software development. We combined the various aspects that occur on a psychosocial level (including involvement), such as participants' attitudes or beliefs, in the human aspects category. We defined the subcategories through a bottom-up approach from the 231 different research aspects named in the studies. The number of studies that researched each aspect indicates that aspect's popularity.

TABLE 2

Overview of aspects of system success.

Measurement of system success	Definition for study	Examples of original measurement	Number of studies
User satisfaction	User's degree of favorableness with respect to the system and the mechanics of interaction	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	50
System use	Frequency of use of the developed system	Intention to use, system impact, system usage, time spent using	18
System quality	Structured set of characteristics such as a system's functional suitability, reliability, usability, performance efficiency, compatibility, security, maintainability, and portability	Accessibility, accuracy, completeness, flexibility, perceived system quality, product success	17
Project in time and budget	Project efficiency and effectiveness in terms of schedule, budget, and work quality	Project success, overall success, process satisfaction, project completion, project performance, project success, successful implementation	8
Ease of use	Degree to which a user expects the target system to be free of effort; also refers to system friendliness and handling in system's use	Perceived impact on work, system friendliness	4
Data quality	Degree to which the characteristics of data satisfy stated and implied needs when used under specified conditions; accuracy, consistency, and availability of data	Appropriateness of format, availability of historical data, data accuracy, data consistency, data sufficiency	1

Measuring System Success

How to measure system success is quite controversial and difficult.[4] In our review, we found that most researchers define it as being whatever aspect they measured in their study—user satisfaction, system use, or system quality.[9] Some papers used existing conceptual models, such as the technology acceptance model,[10] with the aspects "perceived ease of use" and "usefulness." We've seen in our own review that researchers have used various terms for each of the defined measurements—for example, user satisfaction was named "outcome satisfaction," "perceived system usefulness," or "system acceptance." In our review, we included all measurements of system success investigated by the identified papers and categorized them as user satisfaction, system use, system quality, project in time and budget, ease of use, and data quality. Table 2 gives an overview of the different interpretations and measurements of system success used in our review study.

UPI and System Success

We analyzed the correlation data according to the subcategories of development process (mode of development, user participation, user-developer communication) and human aspects (developer's attitude toward user, user involvement, user's beliefs about developers, user's ability in IT projects, user's attitude toward system, and user's intention to use) and the various measurements of system success. As we only used significant correlations, each positive link (that is, a study reporting a positive correlation between an aspect of UPI and a measurement of system success) increases the confidence in there being a positive effect on system success.

Figure 1 shows the number of identified positive and negative links (positive correlation/negative correlation) and the variation of the correlations between the aspects of UPI and system success. The number of identified links gives an indication of the perceived importance of UPI aspects on system success measurement. The variation of the correlations shows diversity in the different findings and indicates contradicting results. For example, the variation of the correlation between the development process category and "project in time and budget"

Studied aspects between user participation and involvement and system success

System success measurements

Aspects of user participation and involvement	User satisfaction	System use	System quality	Data quality	Project in time and budget	Ease of use	Total
Development process	**31/1**	**8/0**	**16/0**	**6/1**	**8/3**	**1/0**	
Variation of correlations	[0.03 – 0.57]	[0.06 – 0.20]	[0.12 – 0.61]	[-0.16 – 0.16]	[-0.47 – 0.51]	[0.32 – 0.32]	
User participation	28/1	8/0	9/0	-	5/3	1/0	55
Mode of development	1/0	-	3/0	6/1	-	-	11
User-developer communication	2/0	-	4/0	-	3/0	-	9
Human aspects	**31/2**	**12/0**	**3/0**	**6/2**	**1/0**	**0/1**	
Variation of correlations	[-0.15 – 0.64]	[0.11 – 0.59]	[0.38 – 0.47]	[-0.17 – 0.21]	[0.27 – 0.27]	[-0.14 – 0.14]	
User involvement	23/1	7/0	3/0	6/2	-	-	42
User's beliefs about developers	3/0	1/0	-	-	1/0	-	5
User's intention to use	-	3/0	-	-	-	0/1	4
Developer's attitude towards user	2/1	-	-	-	-	-	3
User's attitude towards system	1/0	1/0	-	-	-	-	2
User's ability	2/0	-	-	-	-	-	2
Total	65	20	19	15	12	2	

Positive correlations

Negative correlations

x/*y* = Number of positive correlations/number of negative correlations

[min. correlation – max. correlation]

Overall 133 links
Total 92% (123)
positive correlations

FIGURE 1. Links between aspects of user participation and involvement and system success. The graphs of positive and negative correlations to the right and below the main table indicate the sum of identified links and thus summarize how much evidence exists for a positive influence of each aspect of UPI on the system success measurement.

varies from –0.47 to 0.51, which shows that one study revealed that it's harder to keep a project in time and budget and still let users participate, whereas another study discovered that user participation is very beneficial to ensure that a project stays in time and budget. Finally, the graphs of positive and negative correlations to the right and below the main table indicate the sum of identified links and thus summarize how much evidence exists for a positive influence of each aspect of UPI on the system success measurement.

Overall, our meta-analysis revealed that the user participation aspect of the development process category and the user involvement aspects of the human aspects category are the most popular to research. However, other aspects, such as who's in charge of development (mode of development) or the influence on communication between users and developers, are also commonly researched. User satisfaction is by far the most used measurement of system success. Less than a third of studies used system use or system quality, indicating more emphasis on improving software systems from a user perspective than from a quality perspective.

As we can see in Figure 2, 92 percent of the correlations between UPI aspects and system success show a positive correlation, providing evidence of a robust and transferable effect. Thus

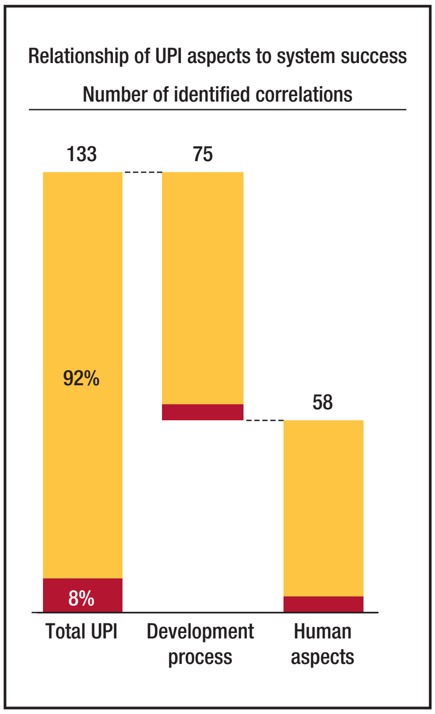

FIGURE 2. Relationship between aspects of user participation and involvement and system success. The 92 percent of correlations between UPI aspects and system success show a positive correlation, providing evidence of a robust and transferable effect.

our study clearly shows that UPI aspects have a positive effect on system success. Of the 133 identified links between UPI aspects and system success, we identified only 10 negative correlations. Six studies reported them, but only one was originally published in the past 10 years.

Our analysis also shows that the correlations among user involvement, participation, and satisfaction are dominant. Users who participate in software development are more satisfied with the system. The same is true for users who are involved, leading us to conclude that user participation and involvement have a positive effect on user satisfaction and system use. Even though the data clearly indicates a positive effect of UPI on system success, some links between UPI aspects and system success have a large variation. There's still no clear conceptual model to measure the effects of UPI, indicating that aspects of UPI are complex to measure and study.

Our review shows that various aspects of users participating or being involved in software development projects have a positive effect on system success. In particular, if users perform activities in the system development process (user participation), such as being the project team leader or being responsible for selecting hardware or software, estimating costs, and requesting funds,[11] the user's satisfaction with the system and therefore its use increases. The same is true if users feel that the system has higher importance and personal relevance to them (user involvement). Thus, we encourage all practitioners to increase user participation and involvement in all phases of their software development projects as much as possible. The positive effects will not only help improve the resulting system from a quality perspective but also increase the system's value for the user.

For the research community, the results of our meta-analysis show that increased UPI leads to increased system success. Thus, research work in the area of UPI in software development is beneficial and should be continued. We've started to develop a first categorization with our categories and subcategories, but we want to encourage other researchers to establish a standard model to enable comparability of future studies. We found that only limited research on the influencing factors of UPI is available, so we encourage researchers to conduct studies in this area. The large variation of the identified correlations calls for more sophisticated empirical studies on the effects of UPI, which would help the research community to further increase confidence in the effect of UPI on system success.

References

1. N. Iivari, "Enculturation of User Involvement in Software Development Organizations: An Interpretive Case Study in the Product Development Context," *Proc. 3rd Nordic Conf. Human-Computer Interaction* (NordiCHI 04), ACM Int'l Conf. Proc. Series, vol. 82, 2004, pp. 287–296.
2. H. Barki and J. Hartwick, "Rethinking the Concept of User Involvement," *Management Information Systems*, vol. 13, no. 1, 1989, pp. 53–63.
3. T. McGill and J. Klobas, "User Developed Application Success: Sources and Effects of Involvement," *Behaviour & Information Technology*, vol. 27, no. 5, 2008, pp. 407–422.
4. M. Harris and H. Weistroffer, "A New Look at the Relationship between User Involvement in Systems Development and System Success Development and System Success," *Comm. Assoc. Information Systems*, vol. 24, no. 1, 2009, pp. 739–756.
5. U. Abelein and B. Paech, "Understanding the Influence of User Participation and Involvement on System Success: A Systematic Mapping Study," to appear in *Empirical Software Eng.*, 2013; http://dx.doi.org/10.1007/s10664-013-9278-4.
6. S. Kujala, "Effective User Involvement in Product Development by Improving the Analysis Of User Needs," *Behaviour & Information Technology*, vol. 27, no. 6, 2008, pp. 457–473.

7. J. Gould and C. Lewis, "Designing for Usability: Key Principles and What Designers Think," *Comm. ACM*, vol. 28, no. 3, 1985, pp. 300–311.
8. D. Dean et al., "Enabling the Effective Involvement of Multiple Users: Methods and Tools for Collaborative Software Engineering," *J. Management Information Systems*, vol. 14, no. 3, 1998, pp. 179–222.
9. M. Hwang and R. Thorn, "The Effect of User Engagement on System Success: A Meta-analytical Integration of Research Findings," *Information & Management*, vol. 35, no. 4, 1999, pp. 229–236.
10. F. Davis, R. Bagozzi, and P. Warshaw, "User Acceptance of Computer Technology: A Comparison of Two Theoretical Models," *Management Science*, vol. 35, no. 8, 1989, pp. 982–1002.
11. J. Hartwick and H. Barki, "Explaining the Role of User Participation in Information System Use," *Management Science*, vol. 40, no. 4, 1994, pp. 440–465.

ULRIKE ABELEIN is a PhD student in the Department of Software Engineering at the University of Heidelberg. Her research focus is on communication between end users and developers in large-scale IT projects. Abelein received an MS in computer science from the Karlsruhe Institute of Technology (formally known as University of Karlsruhe). Contact her at abelein@informatik.uni-heidelberg.de.

HELEN SHARP is professor of software engineering at The Open University. Her research investigates the human and social aspects of software development, focusing on collaboration, teams, and user involvement. Sharp has been active as a conference organizer in the agile software development, object-oriented development, and HCI communities. Contact her at helen.sharp@open.ac.uk.

BARBARA PAECH is a professor of software engineering at the University of Heidelberg. Her teaching and research focuses on methods and processes to ensure quality of software with adequate effort. Paech is particularly active in the area of requirements and rational engineering. Contact her at paech@informatik.uni-heidelberg.de.