

Problems and Challenges of User involvement in Software Development: an Empirical Study

Didar Zowghi

Faculty of Engineering and
Information Technology
University of Technology Sydney
Didar.Zowghi@uts.edu.au

Francesca da Rimini

Faculty of Engineering and
Information Technology
University of Technology Sydney
Francesca.daRimini-1@uts.edu.au

Muneera Bano

Faculty of Engineering and
Information Technology
University of Technology Sydney
Muneera.Bano@uts.edu.au

ABSTRACT

Context: The benefits of involving users in software development projects have been studied extensively in the last four decades and have been reported to contribute to user satisfaction thus leading to system success. However, the relationship between user involvement and system success, being a multi-faceted and complex concept, has introduced many problems and challenges for the practitioners. **Objective:** In this paper we present our findings from a case study to give a deeper understanding of the challenges and problems of user involvement during software development. **Method:** The data in the case study was collected from interviews, observations and project documents. **Results:** We present our results in four main categories related to users, communicative aspects, managerial considerations, and project issues. It was observed that system success is achievable even when there are problems and challenges in involving users. **Conclusion:** Understanding the nature of the problems related to user involvement helps the project managers to develop appropriate strategies for increasing the effectiveness of user involvement.

Categories and Subject Descriptors

D.2 [Software Engineering]: D.2.9 Management

General Terms

Management, Human Factors.

Keywords

User, involvement, participation, software development

1. INTRODUCTION

The success of software development (SD) projects can only be meaningful if the eventual users are satisfied with the functionality and the day-to-day operations of the delivered software. User satisfaction and acceptance has always been considered as one of the fundamental measures of system success [1, 2]. 'User involvement in software development' has been a

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions@acm.org.

EASE '15, April 27 - 29, 2015, Nanjing, China
Copyright 2015 ACM 978-1-4503-3350-4/15/04...\$15.00
<http://dx.doi.org/10.1145/2745802.2745810>

challenging concept both in theory and practice [3]. It is a combination of three different terms that need to be analyzed separately in their individual and distinctive definitions i.e. 1) user, 2) involvement, and 3) software development.

The typical concept of a user is a person who would be actually using the system and facing the changes in the work environment brought about by the new system. However the term 'user' has been considered from many different viewpoints in the literature [4]. In practice, this is largely dependent on the participatory methods and techniques employed in the project [5].

The second term 'involvement' has a clear definition proposed by Barki and Hartwick [6] as 'a subjective psychological state reflecting the importance and personal relevance of a system to the user'. But this term has also been used synonymously with 'participation' and 'engagement'. Barki and Hartwick defined user participation 'as a set of behaviors or activities performed by users in the system development process' [6]. Therefore it is not necessary that the users who are involved in the project should also participate and perform activities.

The third term 'software development' refers to the lifecycle comprising of various stages and activities to build software. It was believed that effective user involvement in the early stages of development is sufficient for capturing users' needs and that their participation may not be required in subsequent phases [7, 8]. But this claim has not been supported widely in the literature [9] and it is observed to be equally important to involve users in design, testing and implementation of systems [5].

With so much conceptual variations and diversity in understanding and implementation of the concept of user involvement in software development, it is inevitable that this gives rise to problems and challenges for practitioners. Literature has pointed out several factors that prevent effective user involvement: e.g. communication challenges, misunderstandings, conflicts, budget, resource, and time constraints [3].

The empirical study reported in this paper is part of our ongoing longitudinal study of user involvement in software development that started in 2012 that complements the findings of our previously published systematic literature review (SLR) on this topic [1, 3]. Our objective of this study was to explore and explain the problems and challenges faced when users are involved in SD in order to offer new insights about this complex and multi-faceted phenomenon. This study is guided by the research question:

What are the problems and challenges faced when users are involved in software development?

We performed a case study of user involvement in software development at an Australian State Government organization. The data for this case study was collected in three phases over a period of two years and in two distinct parts of a specific software development project. We present our results in four main categories: *user related*, *communicative aspects*, *managerial considerations*, and *project related*. The results also show that satisfaction is achievable even when there are problems and challenges in involving users. Understanding the nature of the problems related to user involvement helps the project managers to develop applicable strategies for increasing the effectiveness of user involvement.

2. RESEARCH MOTIVATION

The majority of the studies reported in our SLR [1, 3] have investigated the UI from the viewpoints of the development team alone. The perception and attitude of users about their involvement is equally as important and informative. Therefore, in our case study we have investigated the UI from both developer and user perspectives.

The link between user involvement (UI) and successful system is neither direct nor simple [10]. There are many factors that complicate and influence the successful outcome of any SD project [11]. We have previously identified that 68% of the empirical studies of user involvement have shown positive results, whereas the other 32% reveal either negative or uncertain results [1, 3]. Effectiveness of user involvement is measured by the extent to which the desired benefits and goals of UI can be achieved. It is a double-edged sword that if mismanaged can cause serious problems for all parties concerned [8]. According to our SLR on this topic, the most prominent problems caused by user involvement are related to communications and misunderstanding between users and development teams leading to all kinds of conflicts. Furthermore, the top challenge that hinders user involvement is the lack of motivation, followed by the problems related to user attitudes and behavior.

3. CASE STUDY

We conducted the case study of a software development project in a financial institution of one of the Australian State Government organizations (ASG).¹ One year before the commencement of the project, ASG conducted a large-scale survey among its clients to find out how it could improve its services. This was part of a larger project called the “Client Experience Program” whose main aim was to improve the clients’ experiences with ASG. The result of this survey determined that clients of ASG wanted to use better technology in their transactions with ASG to simplify their work. For example, they expressed the need to replace fax with emails.

The overall IT strategy within ASG was to move to an enterprise-wide Microsoft platform including Outlook. One part of this strategy was to replace the existing Client Information System (CIS), a bespoke Lotus Notes based software, with Microsoft’s Client Relationship Management System (CRM) for the internal users (Clients Relationship Team) within ASG. This was then put onto a client portal so that ASG clients could view their day-to-day transactions through a secure online access. This is mainly because Lotus Notes was to be replaced eventually by Outlook, which mandated the replacement of CIS with CRM.

ASG engaged an External Consulting Company (hereafter referred to as ECC) to tailor Microsoft’s CRM to their needs. The

overall software development methodology was essentially plan-based Waterfall with some prototyping. Although the requirements gathering phase and initial coding had already started when we commenced the case study, design and review workshops were continuing. After a Non-Disclosure Agreement was signed, the data collection commenced.

In this case study, major participants comprised a small team of developers from ASG, the Project Manager from ECC, and the Subject Matter Expert (SME) from the Client Relationship team who was seconded two days a week to participate in the project. Although the organization is relatively small in terms of employee numbers and managerial hierarchy, it is instrumental in the State Government operations. The case study was limited to two sequential SDs (CRM and Client Portal), which were occurring during a period of significant organizational change.

Half way through the project the Technical Project Manager (TPM) from ASG was sacked and subsequently replaced by a new manager who introduced process improvement initiatives in the project, e.g. weekly project meetings that involved all team members including the SME, formal review of all documentations. Monitoring and controlling the project was changed so that the ECC staff reported to the ASG Project Manager. While these practices were introduced in preparation to transitioning into an agile software development approach in future projects, they were also found to be effective in overcoming some of the early challenges of the project.

3.1 Case Study Design and Execution

Case study is an ideal research method to investigate a phenomenon or explain a concept in its real setting [12, 13]. We conducted a holistic explanatory case study to increase our understanding of the problems and challenges of user involvement. The case study spanned two years from January 2012 to December 2013, including the occasionally overlapping data collection and analysis stages. Some significant delays in the project timeline, along with researchers’ availability, impacted the timing of interviews. There were three interview phases: preliminary (January-May 2012), post-installation of CRM software (July 2012-January 2013), and post-installation of Client Portal software (October 2013). Our data set included transcribed interviews with stakeholders, our own observational notes while attending design workshops, and review of formal project documentation. Although the data was collected about many different aspects of user involvement, (for example, communication modes users employed, the nature of business process changes, and how the organization had handled process change management), in this paper we focus only on the problems and challenges captured in our data analysis.

3.2 Data Collection

We conducted 28 interviews with 12 subjects: 5 subjects were interviewed once, 3 interviewed twice, 3 interviewed 3 times, and 1 subject (project manager) was interviewed 8 times. The first author conducted all interviews and the recordings of face-to-face semi-structured interviews were professionally transcribed. Other data gathering activities included informal face-to-face and phone conversations, and follow-up questions and responses by email. Three other data sources were used, namely, observations in various workshops, project documentations, and hand-written notes of phone and informal conversations with some of the relevant ASG employees.

¹ The real name of this organization cannot be mentioned due to ethical consideration and non-disclosure agreement signed by the researchers.

3.2.1 Interviews

Interviewee selection was determined by suggestions from key stakeholders and users within the organization. In the first instance the **Chief Information Officer (CIO)** arranged for the first author to interview the **Technical Project Manager (TPM)**, who then introduced her to **Project Manager (PM)**. The PM selected staff members from different business units to be interviewed. Interviewees included the Principal Consultant of the **external development company (ECC)**, **Business Analysts (BA)**, Client Relationship Managers, Operations Managers, Risk and Compliance Managers, an in-house programmer, and the Head of Client Services who was also the Project Owner and a Steering Committee member. One of the Client Relationship Managers considered as a **Subject Matter Expert (SME)** was assigned by the Head of Client Services to the project team as the business representative with the authority to review and sign off project-related documents. She was interviewed 3 times and played a crucial role both in the SD project and in the information gathering of our case study. As the project unfolded and participants mentioned other important users, the interviewee pool was expanded in consultation with the PM.

Each interview typically lasted between 30-60 minutes. As Hartwick & Barki emphasise, it is important that subjects report “whether or not events have taken place” rather than giving “subjective assessments of an abstract concept (i.e., participation)” [14]. This approach “does not rely on users’ self-assessments of the relative importance of different participation dimensions or different participation activities”. The dimensions of participation and their importance are empirically derived from responses. We followed this approach in our case study. All subjects answered a core set of 20 questions², and the BAs answered an additional 3 questions. Participants described their role in the organization, and reported when and how they had been involved in specific SD activities. These activities included scoping, requirements gathering, design workshops, reviewing specifications, user acceptance testing, training, and cost-implementation evaluation.

Although most of the interview questions required answers based on facts rather than opinions, unlike Hartwick and Barki [14] we considered that people’s evaluation of how the SD and user participation could have been improved was indeed very relevant, and hence elicited some subjective responses as well. A small number of users, in particular the few users who identified themselves, or were identified by others, as “super” or “champion” users, appeared to have no hesitation about sharing their subjective assessments about the project. Some “minor” users also used this opportunity to express their frustrations about perceived communication and project management problems.

The second round of interviews occurred soon after implementation of the CRM system. We devised a new interview instrument focusing on the development process’s final stage and the success of the recently installed software. The third round of interviews required a small number of “key users” to reflect back on the whole project, and to compare the differences in processes and outcomes between the two parts of the project (CRM and Client Portal). In some instances they were also asked to present feedback they had received from their own clients who were external users of the Client Portal. These interview questions were influenced by some of our findings from the SLR.

3.2.2 Observations in Workshops

We observed 4 on-site project workshops. Three were design workshops and one was a “Review Wireframe Workshop”, each lasting 2-4 hours. During the workshops we used an ethnographic approach to data collection. First, we identified the attendees and their organizational role. Second, we took extensive handwritten notes of observed interactions between the workshop presenter and specific users. Third, we wrote up our reflections immediately following each workshop, focusing on the roles specific attendees appeared to play, their individual communication styles, the facilitator’s communication style and presentation methods, and users’ expressions of positive and negative affects about the software development project and how it was being conducted.

In the final stage of our data analysis we reviewed these reflections, identifying whose voices had been the most prominent, which issues they had raised as concerns or criticisms, and whether they had expressed similar concerns during our individual interviews with participants.

3.2.3 Project Documentation

A total of 16 project documents were made available for analysis. We categorised the documentation by organisational author, creating 3 documents classes: those generated by the ECC; those generated by ASG; and a survey document from a major external consulting firm. We reviewed all documents to obtain a more holistic understanding of the project aims, user roles, requirements gathering processes, project management processes, and the internal evaluation of the successes and failures.

3.3 Data Analysis

We used NVivo software application to assist our thematic analysis [15] of the rich qualitative data from multiple sources.

We aggregated the answers to each interview question by interviewees. This was followed by thematic coding of other data sources. While we were analysing the data and coding them further themes emerged that we captured as new nodes in NVivo. The insights we gained primarily arose from iteratively coding the data, and continued reflection upon and discussion about the information in the transcripts among the researchers.

To answer our research question we used two tables from our previously published SLR [3] as a basis for our analysis, namely *problems caused by user involvement* and *challenges of user involvement*. We had generated a collection of relevant node reports from our coding of data in NVivo that we analyzed to look for these problems and challenges. We carefully reviewed all of these node reports with respect to tables from our SLR to ascertain which of them are manifested in this case study. Furthermore, we kept a count of references made to each of these problems and challenges in the transcripts of the interviews. The four key members of the project team who contributed most in identifying the problems and challenges were the PM, BA, SME, and ECC manager. We also examined the transcripts of the rest of interviewees to make sure that they had not mentioned additional issues. But we did not find anything noteworthy that was not already mentioned by the four core members.

Once the analysis of the qualitative data was completed we had a list of 32 reported problems and challenges of user involvement in this case study. Upon further thematic analysis of these identified problems and challenges, we then grouped them into 4 high level categories based on the overall aspect of user involvement that they covered: *User-related aspects*, *communicative aspects*, *managerial considerations*, and *project-related aspects*. The results are presented in Table 1 and are discussed in details below.

² <https://drive.google.com/file/d/0B8AfnfY1xYDd2FVcFJqSGdPSzg/view>

In some cases, perceptions made by one interviewee were echoed by others. In other cases, experiences of involvement or the perception of system success strongly differed amongst participants; this was often related to the user's role in the project. In the presentation of our results we have strived to be faithful to representing agreements and conflicting remarks as stated in the interviews by providing specific quotes from the transcripts.

4. RESULTS

We present the results by providing detailed responses to our research question structured using the four high level categories of problems and challenges that emerged from thematic analysis. We will include the number of times each challenge or problem has been mentioned in the interview transcripts of the four key project personnel (see Table 1). Because of space limitations, in this paper we will only focus on the top problems and challenges that scored >4 in our analysis. It should also be noted that there are overlaps between the problems and challenges mentioned in different categories.

Table 1: Problems Reported by 4 Key Project Personnel

Category	Name	PM	SME	BA	ECC	Total
User-related Aspects	Users' motivation	4	1	1	-	6
	Users' attitudes	-	2	1	-	3
	Users' expectations	2	4	3	1	10
	Users' expertise	4	2	5	2	13
	Users' appreciation of, or resistance to, change	-	-	1	-	1
	Users' cooperation	-	-	-	2	2
	Users' perceptions of the development team	4	-	-	-	4
Total = 39						
Communicative Aspects	Misunderstandings	-	1	3	-	4
	Conflicts	4	2	2	2	10
	Communication skills of users	-	-	-	1	1
	Intergroup hostility	2	2	1	1	6
	Ignoring users' feedback	-	2	-	-	2
	Communication skills of developers and managers	6	6	3	2	17
Total = 40						
Managerial Considerations	Influence on decision-making	-	1	1	-	2
	Lack of management support	3	1	-	4	8
	Change management	-	-	1	-	2
	Degree of involvement	-	-	1	1	2
	Efforts required by users	2	3	3	-	8
	Ineffective user representation & user identification	-	1	-	1	2
	Manipulation of users	3	2	-	-	5
	Imbalance in power relationship	2	-	1	3	6
	Organization-specific issues	-	-	2	-	2
Total = 37						
Project-Related Issues	Time constraints and delays	8	4	2	3	17
	Technical issues & system - complexity	-	5	1	2	8
	Budget	1	-	-	-	1
	Task complexity	2	1	-	-	3
	Project uncertainty	2	-	-	1	3
	Cost & method of user training	1	1	2	2	6
	Staff turn-around & unavailability	6	-	-	3	9
Total = 47						

4.1 User related Aspects

Our analysis identified 7 main classes of problems that are associated to users (Table 1). These include users' motivation (n=6), users' attitudes (n=3), users' expectations (n=10), users' expertise (n=13), users' appreciation of, or resistance to, change (n=1), users' cooperation (n=2), and users' perceptions of the development team (n=4).

4.1.1 Motivation

According to our SLR [8], the top challenge that hinders effective involvement of users is their lack of motivation. Users may not

wish to participate in or be effectively involved in the software development project. In this case study, the SME was fully involved and participated in SDLC, but not all the relevant stakeholders were motivated to participate. For example, the PM stated "key users,...reluctant to get involved", and "Always get stubborn users ...ultimately if people choose to ignore things that are put out and not listen then there's not much we can do". The BA also highlighted: "I don't think the people really took it seriously to review those requirements because during the peer review we didn't really have many comments or feed-back I mean from the Business". On the other hand, when users are motivated, it can have positive impact on their level and degree of involvement, e.g. referring to the extent of her involvement, we heard the SME say: "It's really our system we do feel quite a bit of ownership of this, if we get it wrong it will be our fault".

4.1.2 Expectations

In most software development projects, users may pose unrealistic expectations from the delivered system, especially if the strategic IT decisions are made prior to consultation with, or involvement of, the users. The effective communication of the goals and strategic intent of developing new software or making major changes to an existing one sets the overall benchmark for what the users should expect from the delivered system. Furthermore, looking at users' expectations from a finer-grained point of view, requirements specification document will set the expectations of the users of the end product and are used by them during UAT.

Users also form their expectations about the software development processes based on what is communicated to them by the management at the outset. These expectations include the level and extent of users' involvement in various stages of the SDLC, the degree of their participation in the decision-making processes, and what their relevant duties are. In relation to the delivered software, we heard in many instances from PM, SME and the CIO of ASG that the vendor promised more at the outset than what they actually delivered, hence raising the expectation of ASG staff. The PM stated: "a feeling at the start that the issue of Lotus Notes integration was not made explicit enough by the vendor".

In the CRM project, apart from the SME who spent a substantial amount of time participating in the SD project, other users were not active participants. From the SME's perspective, there was a feeling that expectation of the degree and level of her involvement was not clearly articulated at the outset. In particular, the responsibilities of the ECC manager (who was conducting all the requirements elicitation and design workshops) were not clarified. The SME was dissatisfied with the way the whole process of requirements elicitation was handled by the ECC manager during the workshops for CRM. These workshops did not have clear agendas, no minutes were taken, and no follow up actions were taken. Overall, from her comments we inferred that her expectations from ECC were not met in the CRM project.

The BA believed that Business was not deeply involved enough in UAT for the CRM project and therefore did not get well enough acquainted with the software: "in the UAT it was supposed to be like for the Business, it was the moment of truth for them to really test these new application that we are introducing but if they could have tested it more... then they would have known what the system is capable of doing." She also mentioned the internal continuing disagreement about whether or not the Vendor delivered: "the Business would now say that during the pre-sales engagement...the Vendor were promising things and some of

these things...got stuck in the mind of the Business users but what I can say is that these things were never in the documents."

4.1.3 Expertise

Users may not have the right level of expertise to participate in the SD project. This may result in their feeling of being inadequate or intimidated, hence displaying resistance to engage with the project. There are different kinds of expertise that one would require from involved users. On the one hand, they are expected to have a deep knowledge of the application domain (so-called Subject Matter Experts), and on the other hand, they are also supposed to have a thorough knowledge of any existing software system that will be replaced by the new system. Another layer of skills needed by users that are selected to actively participate is their ability to read and review requirements specifications. These are highly structured and technical documents that in most cases are written primarily for software designers and not for users. As such they may contain structures and highly technical language to describe the behaviour of the software. Users who are selected for active engagement with the SD project do not usually receive specific training about software development methodologies nor how to read and review requirements specifications.

Indeed in this case study it was very clear that the SME did not receive any training or guidance about how to read specifications. Furthermore, as observed by the PM, users expressed dissatisfaction with the CRM development methodology; *"Well we don't know much about it...It's all brand new to us"*. This impacted UAT, creating uncertainty; *"people felt they didn't really know what was going on but also it made...UAT challenging because... they're supposed to be testing it but it's the first time they've seen it."* He further alluded: *"... a more agile approach would certainly have made for a smoother UAT phase and a smoother implementation. I think it would have probably reduced the level of frustration amongst users and ...mistrust and fear of what they're actually getting"*.

The SME in many instances expressed her feelings of inadequacy and being under-skilled for the tasks she was discharged to do in this project: *"I admit I don't have any IT background... skills"; "someone like me who is not a specialist and then has to know, has to sort of learn how to read specs and interpret this";* and a sense of info-overload: *"numerous almost daily meetings (at start)"; "because I am not in IT and I'm not used to all this paperwork that goes with IT."* She also expressed self-doubt related to responsibility of decision-making: *"You just have to make some decisions... so I don't feel I'm doing the wrong thing but I think... I could always do it better"*.

The BA expressed concerns about the level of the required skills by the involved users. She talked about differences amongst ASGs' divisions' ability to contribute ideas in the requirements workshops: *"since we were the proponent of the project, easily we were able to grasp and define these concepts for CRM. But then people from the Business they're lagging behind...it took them a while to comprehend ... what is a client because they view a client in different ways."* Furthermore the BA described limited technical skills of users, and resistance to new product (regarding UAT) places more time demands on BA. Another concern was about varying abilities of users to understand business processes and software interface changes in the topic-specific software training workshops: *"some of them were not able to digest what were discussed during the workshops"*.

Another set of issues raised by the BA which has an overlap with users' attitudes is the inability of the users to offer comments or feedback in the peer review of the specifications which is

manifested in this comment: *"I don't think the people really took it seriously to review those requirements because during the peer review we didn't really have many comments or feed-back"*. She also mentioned lack of employees' understanding about the software's potential uses, and users.

The ECC commented about users' lack of skills and knowledge of software products. *"...they don't understand the feature set of what you can do with these packages."* And *"not a lot of the users have discovered some of the hidden features of the project yet ... some of those things will come out once we decide to go ahead and offer them more training..."*.

4.2 Communicative Aspects

We identified 6 main types of reported problems in communicative aspects. These include misunderstandings (n=4), conflicts (n=10), users' communication skills (n=1), intergroup hostility (n=6), ignoring users' feedback (n=2), and developers' and managers' communication skills (n=17).

4.2.1 Conflicts

Differences of opinion, disagreements about goals, and conflicts about what should have been delivered, can arise in any SD project. Users reported conflicts that arose at various stages of the project life cycle. These conflicts produced delays, generated mistrust and user dissatisfaction, and ultimately propelled management to change project roles, and personnel.

We observed major conflicts between the SME and the ECC manager. According to the ASG PM, during the CRM project the ECC manager *"lost credibility"* and the Business didn't want him on the project. *"They felt that he over-promised and under-delivered...he lied to them"*. When the ECC was *"put on the spot with a lot of questions from the Business [he] just took the easy path out without being explicit to the Business of what that actually meant"*. This conflict resulted in the more powerful ASG stakeholders halting the ECC manager's involvement in the Portal phase. They changed their management processes and removed much authority and decision-making from the ECC.

A disagreement between the PM and Business users on one side, and the TPM and the ECC on the other, delayed UAT by one week until all parties *"could agree what would constitute an acceptable environment"*. This produced some positive outcomes as some specific functions were addressed prior to testing. Furthermore, the internal team considered that finally they could exert some *"explicit"* influence in the CRM project.

There were conflicting perceptions during UAT about software quality. From the ECC's perspective, users identified many defects but these were not major. However, the PM noted more bugs than anticipated, although because people had not seen the system for a few months they were unfamiliar with its navigation and interface. This might have influenced the decision to change user involvement in the subsequent Portal project (adopting the agile practice of stand up meetings to regularly inform users about the development effort).

Nevertheless, these bugs, and a growing dissatisfaction expressed by users about project communication and decision-making, might have increased skepticism about the ECC's motives. For instance, the SME said, *"a lot of the things that we were sold at the beginning of how the system would work and how the system is so easy to integrate don't work"*. However, the BA claimed that although these things got *"stuck in the mind of the Business users"* they were *"never in the documents"*. Here we have a conflict that puts the external developer (i.e., ECC) and some internal stakeholders (i.e., development team) on one side, and

users on another side. Perhaps this could have been mitigated by clearer communication between the BA and users at the outset, especially with regards to ‘decoding’ the specifications for users.

Some user comments implied that the ECC had deceived them for financial gain; once development was handed over to an internal development team this could no longer be the case. In any event, conflicts eroded confidence in the ECC whom once users had considered to be “*very experienced*”.

A discrepancy existed about whether the old system had been fully decommissioned. Although it was claimed that it had been, Business users disagreed because data migration issues forced them to do extra work to install the CRM on time. This increased workload added to user dissatisfaction until the technical issues were resolved. Finally, we observed disagreements between various team members and users about the most effective training methods. For example, the TPM’s preference “*was not to do hands-on training, so just watch and talk*”, requiring the ECC to convince them “*that it’s probably not a good idea*”. We return to training problems later.

4.2.2 Intergroup hostility

Personality clashes between the users and the development team may create hostility in both groups [3]. In the deeper level of discourse analysis that we performed, only 6 instances of intergroup hostility were mentioned. However, when we reviewed the comments by all users, a different picture emerged. This holistic view describes an intergroup hostility in which one group comprising the ECC and the TPM, is increasingly estranged from another group comprising the Business users, with the PM placed uncomfortably in between.

In his final interview, the PM noted user “*frustration*” with the ECC, stating: “*people don’t like the Vendor*”. The SME compared the ECC unfavourably with the internal developers, and questioned their honesty, saying “*I think they promised us the world and then we sort of got a little island on the world. We didn’t bloody get the whole globe*”. The BA tended to side with the ECC, and was certain that the Vendor had ~~delivered~~ on requirements. Finally, the ECC blamed the ASG’s management for delays in the UAT phase because they had not back-filled the SME’s position, and therefore she “*couldn’t get any time to the testing*”. This opinion was confirmed by the SME who had to take care of her day-to-day job as well as being a permanent member of the development team, responsible for signing off the specifications and being fully involved throughout the SDLC.

4.2.3 Developers’ and managers’ communication skills

Our analysis suggests that the most frequent problem cited is related to the communication skills of the project development team and managers. Software development is arguably as much a linguistic activity as it is a technical one. These problems were reported mainly by the PM and the SME. The main criticism of communication skills came from the users, as the PM, SME and BA all indicated that the ECC had not communicated effectively in a number of areas.

Information overload resulted in the main users being “*probably more informed than they’d like at times*”, the PM said. Likewise, the SME described the CRM workshops that the ECC ran as a “*nightmare*”, and “*disorganised and ineffective...repetitive*” meetings caused “*frustration*”.

In contrast, sometimes there was not enough communication. Documentation was sometimes “*incomplete*”. For example, the

post-implementation online help documentation was not comprehensive, causing many help desk calls that then required the BA to create visual procedures.

The ECC’s poor communication skills, lack of planning, and no provision of training materials compounded users’ perception of “*inadequate*” training. “*You get stuck on a subject, you don’t finish what you’re talking about, and you’re sort of just left hanging*”. These problems could have been avoided by organizing the training sessions and materials in advance, the SME suggested. Similarly, the BA considered that the ECC’s “*classroom*” method did not produce a “*real transfer of knowledge*”. Most users found it difficult to understand the software orientation by passively looking at a screen.

Perceptions of poor communication skills ran both ways to some extent. The ECC said that “*sometimes the people on the ground may not be completely aware of the consequences*” of decisions made at the “*top level*”, suggesting that managers are either not communicating decisions clearly, or users are not listening.

The BA noted that the first post-implementation review (PIR) was unstructured and unsatisfactory, only asking what went well or badly. We can confirm this perception from our PIR documentation analysis. Finally, the SME intimated that software methodologies could produce, or exacerbate, poor communication outcomes. For example, the clear shortcomings of Waterfall methodology used in the CRM project influenced the organization’s choice of the hybrid method for the Portal project. “*We had done a lot of pre-work, you know, talk, talk, talk, talk, talk and then all of a sudden we were presented with a product*”. In contrast, the Portal project introduced formal reviews and stand-up meetings where developers showed users prototypes to elicit feedback. The SME reported that when the internal developers explained the problems they foresaw for specific features the Business had requested, she would reconsider the requirement and then tell the developers “*... ‘I don’t exactly need X, you can give me X2’ and they go ‘Oh really because then we don’t have to break these other 10 things to fix this X’*”.

4.3 Managerial Considerations

We identified 9 problem types that users reported which are related to managerial considerations. These include influence on decision-making (n=2), lack of management support (n=8), change management (n=2), degree of involvement (n=2), efforts required by users (n=8), ineffective user representation and user identification (n=2), manipulation of users (n=5), imbalance in power relationship (n=6), and organization-specific issues (n=2).

4.3.1 Lack of management support

The level of management people referred to depended on their own positions, either as employees within a very hierarchically-structured government organization, or, in the case of the ECC, as an external service provider dealing with that same hierarchy. Therefore, we can assume that individual perceptions of the degree of managerial support are influenced both by people’s professional status in relation to the organization, and also by their own power to resolve problems.

The SME’s discussion of the work involved in reviewing specifications, with all the associated “*argie-bargie and negotiation*” while still being required to do her “*normal*” job, suggests that management had not supported her adequately, a problem the ECC also noted. The ECC flagged potential time and resource problems if the Business did not allocate adequate resources to do the job, saying that the ASG realised that the timeframe was “*very short and very aggressive*”. Therefore, he

advised, it was the team manager's responsibility to ensure that the project would not take years to roll out by allocating the SME "full time" to this project. Without such a commitment, a project can be beset by delays and lack of continuity. The ECC recalled that meetings had to be cancelled or replacement attendees found. Importantly, any replacement had to be "empowered" and "the right user ...someone who can come in and make decisions". The SME confirmed that she would sometimes send a junior team member in her place.

Finally, the ECC's complaint of the lack of sufficient hardware for training (in this case 8 laptops) also suggests a lack of management support. However, it could be the case that the ECC had not communicated the resources the ASG needed to commit to training. This is indicated by the PM's observation that the ECC had not thoroughly prepared for training thereby creating an unsatisfactory situation.

4.3.2 Efforts required by users

User involvement requires extra effort which may not be always possible, or easy, or without producing unintended consequences, all of which can impact key employees professionally or personally. Failure to expend extra effort could result in project delays or failure. Notwithstanding some delays, users' commitment to achieving project goals appeared to be very high. Many interviewees credited key users' dedication, with the SME attracting high praise.

The SME's challenge of how to best represent her team to ensure "team decision[s]" required effort from everyone. "I would like to be better at giving...the feedback back to my team and getting the feedback back but it is...time consuming and it's challenging when everyone is doing other work as well". Heavy involvement throughout the SDLC required her to work late nights and weekends "to get the real work done". Managers expect a "salaried professional" to do her normal job even if this encroaches on personal time. The additional effort during the transition period created work-related stress because "you just have to shove more work into your day or work longer".

From the first BA's perspective, during UAT use, limited technical skills coupled with their resistance to the new system tools imposed additional demands on her time, saying "when they cry for help...I have to be there". Because people wanted to use Excel templates, she had to help them to execute the test scenarios. Furthermore, because Business had not been deeply involved in UAT they were unfamiliar with the software, producing flow-on effects later in the SDLC. Unanticipated software problems required that "limitation" would be managed by users somehow. Although most defects were medium to low in terms of severity, some were a system limitation that users had to learn to live with. This implies that users may need to continue to exert extra effort beyond the initial implementation phase to overcome problems associated with unfinished development or while bugs being fixed.

The PM highlighted that it was a management challenge to convince "key stakeholders" why they should participate. An SD project is "quite demanding" and "asks a lot" of people to get involved. The challenge of convincing management and users to participate is commonly faced by software development teams.

4.3.3 Manipulation of users

Manipulation of users include situations in which the development team may try to make them accept a system that might not fully satisfy their requirements. In our case study this accounted for 5 instances management-related problems. In most cases, the ECC

represents the 'management' in these scenarios even though they were not really managing the project but they had certain position of authority.

In regards to deficiency of the fit gap analysis, some functionality problems were at odds with what the ECC had indicated the software could do: "things are...promised that aren't then delivered", and this problem could be seen as the ECC manipulating users into committing to a sale. The seamless integration ECC had promoted did not match the reality, yet "it was all said it was going to be so easy", the SME commented.

4.3.4 Imbalance in power relationship

Asymmetrical power relationships between users and development team may affect project outcomes. For the PM it was a "challenge" to feel that he had enough personal power and authority to ensure that deadlines could be met, or that the project was being adequately resourced. Although it was his job to "ask questions", once he had received a response, "what more can you do?" Furthermore, "if you're relatively young working with two much more senior people, and again they are product experts and they're experienced", then he had no choice but to acquiesce to their authority. The PM also alluded to the powerlessness users had felt about the ECC's competency. Prior to the PIR there was slight "dissatisfaction" about "things that we didn't do in the best possible way". In comparison, once a new team was involved people had a collective sense that "we're moving on, we're doing it a new way". So the PIR had served as "a timely milestone...to get people focussed again and quite optimistic about the next". The Porter project shifted the power relations that had characterised the CRM project.

The first BA focussed on different power asymmetries that existed in the ASG itself. First was a structural issue. Different divisions "work in silos" with their own concepts. This meant that "you have to exert more efforts for people to bring them together and to understand what you're talking [about], from the larger organizational perspective and not from their own world". Months into the project, people "were not understanding each other. Still there were questions and some of these questions were coming from heavy-weights". Presumably these managers could delay decisions until they understood project goals and processes.

Finally, there were the power issues that the ECC mentioned, and these mainly involved ASG managers. For instance, no end users had attended the initial hands-on product training sessions, even though this would be the first project where there was "high-level user involvement". The ECC attributed this to the TPM who had predicted that users would not participate because "they're not going to want to touch it until ...after it's been built". Whereas the ECC thought that UAT is dependent on users "having an opportunity to actually have played with the software...before testing...while it's being developed". A more equal power balance between the ECC and the TPM might have avoided this situation. Delays in UAT were out of the ECC's control, as it was ASG management's responsibility to give the SME time to do testing.

4.4 Projected-related Aspects

Analysis has revealed 7 types of problems that are mainly associated to the software development project and its overall management. These problems were: Time constraints and delays (n=17), technical issues and system complexity (n=8), budget (n=1), task complexity (n=2), project uncertainty (n=2), cost and method of user training (n=6), and staff turn-around and unavailability (n=9).

4.4.1 Time constraints and Delays

One of the factors that have long been considered as an indicator of successful software development projects is delivering the system on time. Involving users in software development projects is always time and resource intensive. It is important to determine the intensity of their involvement for the corresponding desired benefits. So, the relationship between the desired benefits and the degree and level of user involvement should be ideally analyzed at the outset. Also these decisions will have an impact on which stages of SDLC users should be involved.

In this case study, almost all those who were interviewed stated that they encountered delays both in the CRM and the Portal projects. Various reasons were given for delays in different stages of SDLC depending on that person's point of view and their level of involvement. We did not go out to fully explore the reasons for project delay by asking specific questions of all the interviewees about the causes. However, some interviewees themselves provided us with explanations of these issues. What is not clear, however, is whether all of these delays could be directly attributed to user involvement. It is difficult to ascertain from the analysis of the data how many of these delays could have been avoided if the project was managed more effectively or if the processes at ASG were more formal and followed by all the relevant staff.

The PM talked about delays due to architectural reviews and expressed concerns about missed deadlines in getting documents signed off. He mentioned conflict between the business users and the TPM and ECC regarding the UAT start date that resulted in a delay of at least 1 week. Moreover, he mentioned *"resistance"* from business users regarding the length of training sessions which was *"slightly rushed"*. *"Some people would say there were times when they probably felt like it was a bit rushed because we were applying pressure on them to meet certain timelines"*. The PM expressed his views about the causes saying that workshops would have been more effective if scheduled over a shorter period of time, ideally 3-4 days straight, but this was impossible because of users' unavailability. In relation to the TPM's departure, he said that it cost the project 1-2 months of development time. The BA's departure from the ASG right in the middle of writing business requirements for CRM was also costly since time was spent to *"up-skill"* the new BA on the team.

The SME had strong views about time constraints. Several times she flagged her work overload when reviewing specification while still required to do her own normal job. In order to ensure team decision on all matters concerning their requirements, she had to spend additional time giving the feedback to my team and getting their feedback. This is certainly *"time consuming and it's challenging when everyone is doing other work as well"*. Her heavy involvement meant that she had to work *"a lot of late nights and come in a lot of weekends to get the real work done."*

The ECC manager identified other delays and the reasons for them. He also confirmed what we heard from both the PM and SME: *"The timeframe was too aggressive so they [business users] felt they couldn't really commit the amount of time for their work plus the project"*. Unfortunately the users raised concerns about the *"aggressive timeframe"* and complain went to the CIO, which ultimately cost the TPM his job. According to the Operations Risk Team manager, the TPM's departure *"put a spanner in the works"*. The ECC manager in confirming that the CRM project was not delivered on time stated that delays caused by many factors: *"we needed more time for UAT and stretch that out, we need time to training so we added another week. You know just some re-training..."*, and *"certain things had to be redone before we could*

move ahead like the whole integration interface piece, that kind of put a bit of a roadblock in the process."; *"the design phase itself, the whole documentation review process, then getting the business requirement documents signed took forever"*. He blamed ASG processes: *"their processes weren't as effective, in fact they introduced a peer review process recently and they didn't have that, they would just farm out the document to...ten people but never have meetings to review collectively the feedback"*.

In summary, delay in requirements specification document sign off (pre-implementation), last minute changes from the manager of the business unit causing scope creep (during implementation), business users' complaints about unrealistic project timelines that led to the TPM's departure caused over one month of delay in CRM project delivery, and the SME's feedback about readiness delaying UAT (post-implementation). Moreover, it is clear that communicating project delays and the reasons to stakeholders in a timely fashion assists greatly in managing their expectations.

4.4.2 Technical issues and system complexity

It is difficult to involve users (especially those who are not experienced in software development projects) in the development of complex systems. Although the CRM software cannot be considered as a typical large and complex system, there were a number of technical issues at different stages of SDLC (some related to complexity) that were mentioned in the interviews.

One of the technical problems faced by the users was data migration. ASG's move to an enterprise-wide Microsoft platform including Outlook meant that the existing Client Information System (CIS), was replaced by Microsoft's Client Relationship Management System (CRM). Lotus Notes was to be replaced eventually by Outlook. But Outlook migration had not taken place before the CRM project began. In our conversations with the SME, in relation to the migration problem she stated: *"When we heard from providers, suppliers, they may have been talking in an Outlook world and we are not in an Outlook world ... we really get all that integration that we should be getting with this."* She was annoyed with *"clunky"* software where *"you still write in Outlook or in Lotus Notes and you just save to CRM, you don't really write in CRM."* Later on she mentioned lengthy manual data migration process. This caused work flow problems: *"It's not like we've just turned on the new system and were using it. It will take months for us to get the system up and running. We'll be doing it all this financial year..."*. While this was perceived to be a big problem by the SME, the BA dismissed it.

There were also problems with Sharepoint software. There were cascading problems, SME said: *"they had to do a lot of software development to hook SharePoint in to be as easy to use and robust"*. This was echoed by the ECC manager: *"only data migration issues that we ran into was around SharePoint"*.

The last technical issue that was mentioned by ECC manager was about not knowing enough about the legacy backend system in the integration process: *"It's not simple, as they go through the requirements they discovered their back end systems are not exactly the way they thought they were, it was a lot more complicated to get information together. So there was a lot of backwards and forwards around clarification between the IT guys and the requirements"*.

4.4.3 Cost and method of user training

In this category we looked at two aspects of training, first, adequate training for the users who were expected to be involved in the SD project. This training covered the processes and activities that would unfold in the SDLC, for example, how to

read and review requirements specifications. Second, the usual training that user of software receives about how to use the delivered software. Training the users about the nature of their involvement and usage of new system will undoubtedly put additional cost on the project budget. Furthermore, methods that are used by the development team to train the users have to match the skill set and needs of the end users.

In this case study, there were a number of concerns raised about both the cost and method of training for user involvement and end user system training. Firstly, no training was provided for the SME who was a member of the core team and heavily involved in the SD project. She mentioned several times that she has never been involved in software development projects, and consequently she did not know how to read specifications. She said: *"someone like me who is not a specialist and then has to know, has to sort of learn how to read specs and interpret this, what does this really mean in our everyday life, how is this going to change our user experience and our clients' user experience? So that's a challenge for me that I don't know that I am really living up to that well"*. Secondly, she conceded that by increasing her involvement in the Portal project through attending stand-ups she developed a much better understanding of technical feasibility and the difficulties associated with making changes to the software. Overall, training was not provided for users about their involvement in the project.

In regard to post implementation user training, concerns were raised by the PM, BA and SME about the ineffective training provided by the ECC manager. The SME mentioned *"we always seem to either run out of time. He [ECC manager] said that everything was in a PowerPoint which ... I still haven't seen....We're still asking for training"*. And *"we don't have a booklet or any training manual or anything."* That is why they decided, *"With the next phase we will deliver the training because we weren't happy with what we were given"*. The BA also concurred. She was dissatisfied with the ECC's training method as it did not produce a *"real transfer of knowledge"*: *"the orientation, it was just in a classroom type" "people would have difficulty digesting that orientation unless you are really really good ... that you can by just ... looking at the screen you would be able to digest it."* She advised that these problems had generated the need for more training, ideally not *"group"* training but on a *"person-to-person basis"*, which would have implications for staff resources. The ECC manager on the hand mentioned disagreement between him and the TPM regarding most effective method of training: *"His [TPM] preference was not to do hands-on training, so just watch and talk and then I convinced them that it's probably not a good idea."*

4.4.4 Staff turn-around and unavailability

One of the unique attributes of this case study that we did not encounter in our SLR was the unexpected staff turnaround during the project. We also came across several instances of staff unavailability due to being on various forms of leave that caused delays in the project and challenged the involvement of the users.

The PM described a number of these instances. First the CIO was away which held up the signing off on the solutions, which in turn delayed UAT. It is not clear why they could not have an acting CIO in her absence. Then the timing of key project events was delayed because 5 people were on leave including during the PIR. The ECC manager also alluded to these forms of delays that affected the degree of user involvement: *"We had one individual that's been sick unfortunately, can't do much about that."*. And *"...should we [ASG] send out version A [of business requirement documents] to five people, there was no deadline... Somebody*

might be on holidays for 3 weeks". *"Business requirement sign off process was way longer than they needed to be"*.

5. DISCUSSION

Our investigation of these two distinct parts of a large SD project (CRM and The Client Portal) occurred during a period of significant organisational change. There was 'key staff turnover' at 'critical times during the project' and 'significant changes to business processes across many teams in front/back-office' were made as existing software was replaced.

Four distinct categories of problems and challenges emerged from our analysis that provided a structure to summaries and discuss the results. In each of these 4 categories, two problems and challenges have scored much higher than others; bringing the total to 8 top most challenges of user involvement. In *user related aspects*: expertise=13, expectation=10, in *communicative aspects*: conflicts=10, communication skills of developers and managers=8, in *managerial considerations*: lack of support=8, effort required=8, and in *project related*: time constraints and delays= 17 and staff turn-around=9.

We can divide the overall user involvement into three separate stages of SDLC: pre-implementation (participation in all requirements elicitation, analysis and design related activities, negotiation and prioritization of requirements), during implementation (activities such as review of prototypes, providing feedback about the development processes, attending standups and project meetings), and post-implementation (UAT, training, raising change requests, attending the post implementation review meetings and to resolve outstanding issues). We can now summarize and discuss the challenges and problems related to each of these stages.

In pre-implementation, users were expected to be heavily engaged with the project team to develop, negotiate, and agree on the specification. Our results confirm that all of the top 8 challenges were faced during the CRM stage of the SD project, perhaps the hardest ones related to the communications skills and expertise of all parties concerned, followed by the efforts required by the users, unfamiliar with software development, to make themselves available and take appropriate actions in a timely manner. Many of these challenges were addressed in the Portal stage that had a positive impact on both user engagement and satisfaction.

During implementation, for the CRM project, which used a plan based software development and mainly controlled by the external consultants, there was hardly any user involvement. This caused many problems that ultimately led to ASG management making drastic decisions of changing their processes and management. During the Portal phase of the SD project, ASG introduced daily standups and formal review processes. Control of the project was transferred to ASG and the ECC's involvement in decision making was significantly reduced, turning the project into a somewhat agile and fully in-house development. Users were actively involved in the Portal project resulting in satisfied users.

In post-implementation, once again to some extent all 8 top challenges were faced. In the CRM stage of project, UAT was delayed, training was unsatisfactory, PIR was ineffective and users overall were raising many issues and concerns. On the other hand, because of the changes made by ASG management to the software development practices, many of these issues were resolved in the Portal part of the SD project. The SME and BA played crucial roles in the project and their skills, motivation and attitude was paramount in making the project a success even though there were delays and the cost was over budget.

6. LIMITATIONS

The main limitation of the single case study method in the UI field is that due to context-specific variables (including organization size and type, project complexity, software development methodology, number and diversity of interview subjects, size of data set, etcetera), it can be difficult to generalise the findings emerging from the data analysis [12, 13]. Nevertheless, we consider that this case revealed attitudes, behaviors, problems and some solutions that could be of relevance to other SD projects. We have also conducted a second case study, in a different but comparable ASG. In 2015 we intend to perform a deeper analysis of its data set using the identical categories of analysis as this first study, but considering other dimensions of UI, thus giving us more evidence to support any generalisations and conclusions.

Furthermore, by identifying and describing the relevant variables in this case, and giving a fair degree of contextual information about the organization and its 'big picture' aspirations with regards to the SD project, we consider that others engaged in, or planning to conduct, similar qualitative research projects will find that this case study offers a useful comparative example to their own cases.

Our methodology employed rigorous, repeatable data collection techniques, and we ensured that we interviewed key personnel more than once using the semi-structured format that allowed us to incorporate significant new information in the questions. We employed iterative data analysis steps, including fact checking where necessary. We subsequently chose the 4 most significant project roles on which to perform the final iteration of transcript analysis. All stages of the research were accompanied by ongoing dialogue amongst the 3 researchers about the emergent findings, and how they related to the SLR.

7. CONCLUSION AND FUTURE WORK

We have presented a comprehensive analysis of rich qualitative data collected from an explanatory holistic case study of users' involvement in a software development project. Our objective was to identify the challenges and problems faced in UI by all parties concerned and compare our findings with the literature of the last 3 decades. Understanding the nature of these challenges and problems can assist project managers to develop more effective strategies for increasing the effectiveness of UI. UI does not begin with scoping and requirements elicitation, then disappear, only to reappear at UAT and again in training. Rather UI should be conceived as a continuum, a wave, not a particle.

Hence, development teams must give careful consideration to the timing and extent of user participation at different stages of SDLC. In particular, it is paramount to recognize the specific point in time when UI is absolutely crucial and appreciate that involvement in one stage (pre-implementation) is not necessary more effective or necessary than another stage (post-implementation). Furthermore, specific attention should be paid to users' ability to conceptualize the impact of the system on their work practices, which can only become clear once the users have experienced the system.

A major part of managing UI is maintaining their interest throughout the SDLC by: Communicating project decisions effectively and in timely manner, Informing users of any project delays and the reasons for the delays, and Continuing to give them every opportunity to review the system artifacts under construction, receive their feedback and act upon their comments. It is also imperative that managers back-fill key users' "normal"

positions, to avoid them from having to do essentially two jobs if they are to participate effectively in the SD project. The success of any SD project should not depend on users risking burning out personally to achieve project objectives, when proper planning could easily mitigate this risk.

For future works, we are currently in the process of analyzing the complete data set of this case study taking into account other dimensions of user involvement as identified in the literature. We have already replicated this study in another organization in Australia for comparative analysis. Based on the findings of the SLR and these two case studies we plan to develop a framework for the effective management of user involvement in software development.

8. ACKNOWLEDGMENTS

We would like to express our gratitude to ASG management and staff for allowing us to conduct this study. This research was partially funded by the Centre for Human centered Technology Design Research. Muneera Bano is the recipient of the Faculty For The Future (FFTF) award from Schlumberger.

REFERENCES

- [1] M. Bano and D. Zowghi, "User involvement in software development and system success: a systematic literature review," in *17th International Conference on Evaluation and Assessment in Software Engineering*, EASE 2013, pp. 125-130.
- [2] U. Abelein and B. Paech, "Understanding the Influence of User Participation and Involvement on System Success—a Systematic Mapping Study," *Empirical Software Engineering*, pp. 1-54, 2014.
- [3] M. Bano and D. Zowghi, "A systematic review on the relationship between user involvement and system success," *Information and Software Technology*, vol. 58, pp. 148-169, 2/ 2015.
- [4] J. Iivari, H. Isomäki, and S. Pekkola, "The user—the great unknown of systems development: reasons, forms, challenges, experiences and intellectual contributions of user involvement," *Information systems journal*, vol. 20, pp. 109-117, 2010.
- [5] E. Carmel, R. D. Whitaker, and J. F. George, "PD and joint application design: a transatlantic comparison," *Communications of the ACM*, vol. 36, pp. 40-48, 1993.
- [6] H. Barki and J. Hartwick, "Rethinking the concept of user involvement," *MIS quarterly*, pp. 53-63, 1989.
- [7] K. El Emam, S. Quintin, and N. H. Madhavji, "User participation in the requirements engineering process: An empirical study," *Requirements engineering*, vol. 1, pp. 4-26, 1996.
- [8] S. Kujala, "User involvement: a review of the benefits and challenges," *Behaviour & information technology*, vol. 22, pp. 1-16, 2003.
- [9] M. Bano and D. Zowghi, "Users' involvement in requirements engineering and system success," *IEEE Third International Workshop on Empirical Requirements Engineering* 2013, pp. 24-31.
- [10] A. L. Cavaye, "User participation in system development revisited," *Information & Management*, vol. 28, pp. 311-323, 1995.
- [11] J. He and W. King, "The Role of User Participation in Information Systems Development: Implications from a Meta-Analysis," *J. of Management Information Systems*, vol. 25, pp. 301-331, 2008.
- [12] R. Yin, "K.(2003). Case study research: Design and methods," *Sage Publications, Inc*, vol. 5, p. 11, 2003.
- [13] P. Runeson and M. Höst, "Guidelines for conducting and reporting case study research in software engineering," *Empirical Software Engineering*, vol. 14, pp. 131-164, 2009.
- [14] H. Barki and J. Hartwick, "Interpersonal conflict and its management in information system development," *MIS Quarterly*, pp. 195-228, 2001.
- [15] D. S. Cruzes and T. Dyba, "Recommended steps for thematic synthesis in software engineering," *International Symposium on Empirical Software Engineering and Measurement (ESEM)* 2011, pp. 275-284.