Information and Software Technology 58 (2015) 148–169

Contents lists available at ScienceDirect

Information and Software Technology

journal homepage: www.elsevier.com/locate/infsof

A systematic review on the relationship between user involvement and system success

Muneera Bano *, Didar Zowghi

Research Center for Human Centered Technology Design, Faculty of Engineering and Information Technology, University of Technology, Sydney, Australia

ARTICLE INFO

Article history:
Received 29 August 2013
Received in revised form 18 June 2014
Accepted 18 June 2014
Available online 27 June 2014

Keywords:
User involvement
Software development
Systematic Literature Review

ABSTRACT

Context: For more than four decades it has been intuitively accepted that user involvement (UI) during system development lifecycle leads to system success. However when the researchers have evaluated the user involvement and system success (UI-SS) relationship empirically, the results were not always positive.

Objective: Our objective was to explore the UI-SS relationship by synthesizing the results of all the studies that have empirically investigated this complex phenomenon.

Method: We performed a Systematic Literature Review (SLR) following the steps provided in the guidelines of Evidence Based Software Engineering. From the resulting studies we extracted data to answer our 9 research questions related to the UI-SS relationship, identification of users, perspectives of UI, benefits, problems and challenges of UI, degree and level of UI, relevance of stages of software development lifecycle (SDLC) and the research method employed on the UI-SS relationship.

Results: Our systematic review resulted in selecting 87 empirical studies published during the period 1980–2012. Among 87 studies reviewed, 52 reported that UI positively contributes to system success, 12 suggested a negative contribution and 23 were uncertain. The UI-SS relationship is neither direct nor binary, and there are various confounding factors that play their role. The identification of users, their degree/level of involvement, stage of SDLC for UI, and choice of research method have been claimed to have impact on the UI-SS relationship. However, there is not sufficient empirical evidence available to support these claims.

Conclusion: Our results have revealed that UI does contribute positively to system success. But it is a double edged sword and if not managed carefully it may cause more problems than benefits. Based on the analysis of 87 studies, we were able to identify factors for effective management of UI alluding to the causes for inconsistency in the results of published literature.

Contents

* Corresponding author. Tel.: +61 295141860.
E-mail addresses: Muneera.Bano@student.uts.edu.au (M. Bano), Didar.Zowghi@ uts.edu.au (D. Zowghi).

http://dx.doi.org/10.1016/j.infsof.2014.06.011
0950-5849/

1. Introduction

Since the late 70s, it is believed that user involvement in system development ensures system success [1–3]. The idea can be traced to organizational management research, including group problem solving, interpersonal communication and individual motivation [1]. The satisfaction and acceptance of the system by those who will ultimately use it, is considered as a critical success factor for the project [4–6]. There have been numerous studies that have supported this concept (e.g. [1–3,7,8]). Users typically have significant knowledge of the application domain, the tasks they perform, work practices, context of the system use and their behavior and preferences. This form of knowledge is often tacit in nature and thus difficult to be articulated with typical elicitation techniques. User involvement in Systems Development Life Cycle (SDLC) facilitates understanding of their work environment and can improve the quality, accuracy and completeness of their requirements [1,7,9].

Various methods and techniques have been proposed that provide solutions for effective user involvement. Agile methods (e.g. extreme programming), Joint Application Development (JAD), Effective Technical and Human Interaction with Computer based Systems (ETHICS) are examples of the well known techniques [10]. A few recent initiatives involve taking users' feedback from web repositories for development of modern day applications, e.g. for online mobile applications [11], distributed collaborative application development environment [12], software requirements evolution [13], and in service oriented domain [41].

Upon closer analysis, various instances of disagreements have been observed between the authors of the voluminous empirical literature on the topic [1,2]. The conflicts in the results are claimed to be due to the inconsistencies in research method designs [1,2], confounding effects of usage of the terms "user involvement" and "user participation" [2,14,15], and other contingency factors [16]. The major cause among all of them is considered to be the lack of common understanding of the concepts and philosophies of user involvement [1,2]. "***User involvement*** in ***software development*** and ***system success***" is an intricate and labyrinthine combination of three different concepts that need to be analyzed separately in their individual and distinctive definitions.

First, the concepts related to the term "Users", are not considered harmoniously in all the empirical studies [17]. Users play various types of roles in organization. The typical understanding of a user is someone who would be actually using the system and her/his work and environment in some way would be effected by the system. But defining the "user" for a project depends on the participatory methods and techniques adopted during the project. For example, Participatory Design (PD) community defines users as "*the operational workers who are affected by the system, this does not include the manager*", but in Joint Application Development (JAD), users are "*any non IS/non technical individuals in the organization who are affected by the system, this includes managers*" [18].

Second, "Involvement" is used inconsistently in literature as a synonym for "participation" and "engagement". The first clear distinction between user involvement and user participation was given by Barki and Hartwick [14]. They defined user involvement as "*a subjective psychological state reflecting the importance and personal relevance of a system to the user*" and user participation "*a set of behaviors or activities performed by users in the system development process*". Therefore it is not necessary that the users who are involved in the project should also participate and perform activities. Whereas "user engagement" has been used synonymously in the literature as an additional term to both concepts of involvement and participation [8].

Third, "Software Development" is a life cycle that comprises of various phases, includes many activities and is affected by various dynamic and progressive factors such as methodologies used, application domains where software will be situated, and technological changes [2]. It is widely believed that involving users during early phases of development like requirements elicitation contributes most to accurately capturing their needs [7,9]. But it is also important to involve users in other stages of the SDLC, such as design and testing, when these requirements are transformed into technical solutions [18]. In different phases of SDLC various types and levels of participation of users are required. For example, senior management may be required to be involved throughout development, and middle management and other employees (such as Subject Matter Experts), would be required for their contribution during problem identification, requirements elicitation, design and testing [2]. Uncertainty, system and project complexity are important contributors that determine the phases of SDLC for user