

Chapter 7

Operationalizing the Concept of Success in Software Engineering Projects

Marko Ikonen

University of Helsinki, Finland

Pekka Abrahamsson

University of Helsinki, Finland

ABSTRACT

Success in industrial software development projects is critical in the sense of economic survival for companies acting as software suppliers. If the structure or decision-making mechanism is heavyweight, the suppliers will encounter problems when improving their performance due to their inadequate ability to change. This, however, offers an opportunity for more flexible organizations that are able to proactively survive in a volatile software business environment. Taking such an advantage, regardless, requires understanding the components of success holistically. Research on project success regarding software engineering is still fragmented and focuses on the isolated relationships of success. This article operationalizes the concept of project success based on a literature survey and an empirical validation. As a result, an indicative project success model for software engineering projects is proposed. The resulting model is evaluated empirically in a large multinational software corporation setting. The results show the model provides a valuable tool for KIBS organizations to increase their capabilities in running successful projects as well as to find targets for improvements in these projects.

1. INTRODUCTION

Many organizations providing knowledge intensive business services (KIBS; see Miles et al. 1995) in the ICT sector are software suppliers. In order to survive in the tight competition and also gain better profit, these organizations are needed

to provide better customer value than their competitors. This in turn means heavy investments in continuous software development. However, the practice has shown that often these investments have not paid off: anecdotal evidence suggests that half of the ICT projects fail (Ernest-Jones, 2007; Hartmann, 2004).

DOI: 10.4018/978-1-4666-2934-9.ch007

With the term ‘project’ we refer to the definition of PMI (2008): “A project is a temporary endeavor undertaken to create a unique product, service, or result.” In order to make projects more successful, the organizations must realize in what sense their projects already are successful and in what sense not. Otherwise, project-related targets for improvement are difficult to even find. In other words, success needs to be conceptualized by figuring out the relevant elements of it. The goal of the improvements is to reach increased customer-value, decreased project costs, and win-win situations between customers and suppliers, which, in its turn, increases the competence of KIBS organizations.

The literature of software engineering projects has focused on identifying isolated success factors even though the issues behind the success lie on multidimensional constructions with multidisciplinary factors (Ikonen & Abrahamsson, 2010). While the term of project success is understood in a variety of ways (Agarwal & Rathor, 2006), some definitions taking the customer into account can be agreed to have an established position. Examples of such definitions are the four dimensions of project success of Shenhar et al. (2001) and a definition for success by PMI (2008).

The organizations of software development have not had a standard framework to follow systematically, as Section 2 demonstrates. While the concept of success is often overlooked, the concept of project management is equally little addressed theoretically. Koskela and Howell (2002) argue that there is no explicit theory for project management but rather that the theoretical foundation for project management has emerged from the works of the Project Management Institute in their PMBOK¹ guide. As a resolution, Koskela and Howell (2002) propose a novel theory of project, which portrays a project by a transformation view on operations. They suggest that in the transformation view, a project is conceptualized as a transformation of inputs to outputs. A project,

then, is managed with a number of principles. As one of the guiding principles, Koskela and Howell (2002) identify the principle of decomposing, which aims at decomposing the total transformation hierarchically into smaller transformations, tasks, and minimizing the cost of each task independently. From the software viewpoint, this is not a novel realization. In the most basic software engineering textbooks (Sommerville, 2007), a software project is viewed conventionally as a problem of decomposition. The challenge that the software project brings into the project concept is the significant role of the human in the process of software development (Boehm, 1981). Moreover, transformation processes in software engineering are not tangible artifacts as in other engineering fields. Cockburn (2002) suggests that what flows in software development is a series of invalidated decisions, which become validated only after they have been implemented. Thereby, it can be suggested that any model aiming at depicting a software project success model should take these characteristics into account.

This article conceptualizes the project success of software engineering by suggesting a success model which organizes the fragmented area of the literature. It is claimed that this model explains project success and helps practitioners to find targets for improvement in projects so they can be developed. Thereby, it is suggested that the model provides a valuable tool for the KIBS organizations to improve their own projects and to consult their customers, as well.

The model building strategy opted for in this study is characterized by an inclusion of a wider array of literature base than usually expected. We do not raise any particular model above others but rather treat the existing models as equals and compile a comprehensive model as the research model. However, we use the framework of Bertelsen and Koskela (2003) regarding three settings related to project forces. We collate the fragmented area of project success regarding software engineering and

26 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

www.igi-global.com/chapter/operationalizing-concept-success-software-engineering/74139

Related Content

Two Approaches in Assessing Business Continuity Management Attitudes in the Organizational Context

M. Spremic, L. Turuljaand N. Bajgoric (2018). *Always-On Enterprise Information Systems for Modern Organizations* (pp. 159-183).

www.irma-international.org/chapter/two-approaches-in-assessing-business-continuity-management-attitudes-in-the-organizational-context/192978

Business Competence and Acumen of Information Technology Professionals

Gregory Gleghorn (2015). *Technology, Innovation, and Enterprise Transformation* (pp. 302-312).

www.irma-international.org/chapter/business-competence-and-acumen-of-information-technology-professionals/116973

Assimilation of Inter-Organizational Information Systems: Insight from Change Resistance Theory in Public Electronic Procurement

Kishor Vaidya, John Campbell, Jeffrey Soarand Scott Gardner (2012). *Inter-Organizational Information Systems and Business Management: Theories for Researchers* (pp. 18-32).

www.irma-international.org/chapter/assimilation-inter-organizational-information-systems/61603

E-Readiness in IT/IS Implementation: A Benchmarking Analysis Based on 100 Case Studies

Ayman Altameem and Mohamed Zairi (2012). *Cases on E-Readiness and Information Systems Management in Organizations: Tools for Maximizing Strategic Alignment* (pp. 222-254).

www.irma-international.org/chapter/readiness-implementation-benchmarking-analysis-based/61103

Information Technology Infrastructure for Inter-Organizational Systems

Sean B. Eom and Choong Kwon Lee (2005). *Inter-Organizational Information Systems in the Internet Age* (pp. 76-98).

www.irma-international.org/chapter/information-technology-infrastructure-inter-organizational/24488