



# Software Development Methodologies in Organizations: Field Investigation of Use, Acceptance, and Application

*Charles J. Kacmar, University of Alabama, USA*

*Denise J. McManus, University of Alabama, USA*

*Evan W. Duggan, University of the West Indies, Jamaica*

*Joanne E. Hale, University of Alabama, USA*

*David P. Hale, University of Alabama, USA*

---

## ABSTRACT

*The theories of social exchange, task-technology fit, and technology acceptance are utilized in a field study of software development methodologies. This investigation includes the effects of user experiences on perceptions of acceptance and usage of a methodology. More specifically, perceptions of the outputs and deliverables from a methodology and perceptions of challenges and obstacles to using and applying a methodology were found to significantly and positively influence perceived usefulness and negatively influence ease of use of a methodology, respectively, within a developer's organization. Perceived usefulness was a positive and strong antecedent to perceptions of fit between the methodology and client problems, and the strengthening of efficacy beliefs about the methodology. [Article copies are available for purchase from InfoSci-on-Demand.com]*

*Keywords:*    *Information Theory; IS Development Methods and Tools; IS Development Strategies; IS Life Cycle Activities; IS Management Issues; IS Training and Development; Software Development Methodologies*

---

## INTRODUCTION

Over budget, delayed delivery, off-target functionality, poor quality, and lack of business value are some of the phrases that have

been used to describe software project misses or failures (Linberg, 1999; Venugopal, 2005; Wognum, Krabbendam, Buhl, Ma, & Kenett, 2004). Many reasons have been attributed to project shortcomings, including inappropriate

management techniques (Pattit & Wilemon, 2005), inconsistency between the development and deployment environments (MacCormack & Verganti, 2003), and questionable architectural choices (Maranzano et al, 2005). This article investigates another possible explanation for software project failures; namely, the developer's ability or inability to apply a software development methodology (SDM), as well as his/her acceptance of an organization's SDM toward the solution of client problems.

The motivation for this investigation stems from challenges encountered when collaborating and consulting with information technology (IT) professionals and attempting to apply methodologies to projects that have ranged from brainstorming or requirements-only to full scale implementations. Prior research has shown that an organization may use different SDM's in an attempt to improve productivity and reduce costs, sometimes on a per project basis, but this strategy may result in developer dissatisfaction (Avison & Fitzgerald, 2003; Fitzgerald, 1998). Methods such as extreme programming (XP) or paired programming teams have evolved to address some of these issues (McCormick, 2001). These methods require developers to work more closely (i.e., in pairs) and incrementally (McCormick, 2001), changing the nature of the development team by decreasing team size while at the same time increasing communication within and across teams in order to ensure that more than one individual has full knowledge of the solution strategy.

The focus of this article is to view a methodology as a *technology* that can be accepted or rejected, similar in principle to how users can accept, reject, or adopt information systems (Davis, 1993; Venkatesh, Morris, Davis, & Davis, 2003). This perspective builds on the work of Riemenschneider and colleagues (Hardgrave, Davis, & Riemenschneider, 2003; Riemenschneider & Hardgrave, 2001; Riemenschneider, Hardgrave, & Davis, 2002) that broadened the scope of the technology acceptance model (TAM; Davis, 1986) by applying TAM to study the ease-of-use and usefulness of software

development processes and methods. This article extends prior work in three ways: first, by presenting and testing a new antecedent of TAM related to a developer's perceptions of the challenges and obstacles when using a methodology; second, by investigating developer perceptions of task-technology fit between the methodology and typical client problems; and third, by assessing a developer's efficacy of using a methodology to solve client problems within an organization. The investigation proceeds with a review of relevant literature; presentation of the background theories and model for the research; details of the data sample and sampling procedure; analysis of the constructs; and, finally, model testing, results, and discussion.

## THEORY DEVELOPMENT

### Definition

Various terms are used to denote the processes in which software is developed in organizations, including, among others, *method*, *methodology*, *software development process*, *software development methodology*, *software methodology*, *software process*, *software life cycle*, *systems development life cycle*, *systems life cycle*, and *application development methodology* (Anonymous, 2008; Riemenschneider & Hardgrave, 2001; Riemenschneider et al., 2002; Wikipedia, 2007). This article will treat these terms as synonymous and refer to them using the term *methodology* as identified and defined as a "comprehensive guide to developing a system" by Riemenschneider et al. (2002, p. 1136).

### Literature Review

Limited studies have addressed the factors that influence software developer use of methodologies (Riemenschneider & Hardgrave, 2001). Of the various factors that have been studied, Roberts and colleagues (Roberts, Gibson, & Fields, 1997; Roberts, Leigh, Purvis, & Parzinger, 2001) identified five perceptions that can

22 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/article/software-development-methodologies-organizations/1363](http://www.igi-global.com/article/software-development-methodologies-organizations/1363)

## Related Content

---

### Data System-Embedded Guidance Significantly Improves Data Analyses: When Data Is Made 'Over-the-Counter' for Users

Jenny Grant Rankin (2016). *Handbook of Research on Innovations in Information Retrieval, Analysis, and Management* (pp. 125-151).

[www.irma-international.org/chapter/data-system-embedded-guidance-significantly-improves-data-analyses/137477](http://www.irma-international.org/chapter/data-system-embedded-guidance-significantly-improves-data-analyses/137477)

### Agile Knowledge-Based E-Government Supported By Sake System

Andrea Ko, Barna Kovács and András Gábor (2011). *Journal of Cases on Information Technology* (pp. 1-20).

[www.irma-international.org/article/agile-knowledge-based-government-supported/56306](http://www.irma-international.org/article/agile-knowledge-based-government-supported/56306)

### A Model of the Motivation for IT Retraining

Sherry D. Ryan (1999). *Information Resources Management Journal* (pp. 24-32).

[www.irma-international.org/article/model-motivation-retraining/51072](http://www.irma-international.org/article/model-motivation-retraining/51072)

### Strategies for the Cooperation of Higher Education Institutions in ICT

Juha Kettunen (2008). *Information Communication Technologies: Concepts, Methodologies, Tools, and Applications* (pp. 2104-2117).

[www.irma-international.org/chapter/strategies-cooperation-higher-education-institutions/22803](http://www.irma-international.org/chapter/strategies-cooperation-higher-education-institutions/22803)

### Reorganizing Information Technology Services in an Academic Environment

Marcy Kittner and Craig Van Slyke (2000). *Annals of Cases on Information Technology: Applications and Management in Organizations* (pp. 124-147).

[www.irma-international.org/article/reorganizing-information-technology-services-academic/44632](http://www.irma-international.org/article/reorganizing-information-technology-services-academic/44632)