

Chapter 10

An Empirical Investigation of the Perceived Benefits of Agile Methodologies Using an Innovation–Theoretical Model

Nancy A. Bonner

McLane College of Business, University of Mary Hardin-Baylor, Belton, USA

Nisha Kulangara

Palumbo Donahue School of Business, Duquesne University, Pittsburgh, USA

Sridhar Nerur

*Department of Information Systems and Operations Management, University of Texas Arlington,
Arlington, USA*

James. T. C. Teng

*Department of Information Systems and Operations Management, University of Texas Arlington,
Arlington, USA*

ABSTRACT

There is little doubt that agile software development (ASD) methods have gained widespread acceptance in industry. Despite the attention these methods have received, there is little empirical affirmation of the benefits that accrue to those who use agile methodologies. Grounded in the conceptual foundations of innovation diffusion and agile philosophy of development, the authors' study validates a model to assess the perceived advantage of an iterative approach to software development. Consistent with their predictions, the results suggest that evolutionary development - the cornerstone of agile development – is perceived to be less complex and more compatible with the work habits of developers. Further, the findings support the hitherto unsubstantiated claim that iterative development yields benefits to software developers. However, process flexibility, yet another important characteristic of agile development, had no significant impact on complexity, compatibility, and relative advantage. The implications of the study for academics and practitioners, and directions for future research are discussed.

DOI: 10.4018/978-1-7998-3016-0.ch010

INTRODUCTION

The last ten years or so have witnessed a proliferation of best practices that are believed to mitigate the problems that have plagued the software development field. Foremost among these are a set of practices that have evolved from principles articulated in the Agile Manifesto (Agile Alliance, 2001). The key underlying ideas set forth in the manifesto include the need to (example, (Highsmith, 2002)): a) recognize that people and their collaborations should be privileged to flexible processes; b) understand that tangible working software is more valued than copious documentation; c) involve stakeholders throughout the development process rather than simply contracting with them to develop software that satisfies requirements identified upfront; and d) acknowledge the inevitability of change and find ways to take advantage of it. These principles have been catalytic in spawning a host of agile methods that are increasingly gaining traction in industry (VersionOne, 2015; Ambler, 2007).

Traditional software development methodologies followed a linear sequence of phases, with considerable time, money and resources being expended on upfront analysis and design of the system under development. In general, these approaches used detailed and elaborate planning, were characterized by rigid processes, and scarcely considered the notions of continually building and refining working software based on frequent feedback and reflection. In contrast to traditional software development methodologies, agile methods assume that the business world is turbulent, and its fast-changing requirements resemble an ecosystem in which living things must adapt to constantly changing surroundings and situations (Highsmith, 2002). Recognizing this, agile methods embrace a flexible development process whereby working software of value is progressively developed in short iterations, the end of each providing an opportunity to solicit feedback from customers, to reflect on what worked and what didn't, and to reprioritize requirements. Boehm & Turner (2003, p. 17) succinctly describes an agile method as follows: "A truly agile method must include all of the following attributes: iterative (several cycles), incremental (not deliver the entire product at once), self-organizing (teams determine the best way to handle work), and emergence (processes, principles, work structures are recognized during the project rather than predetermined)."

While there appears to be ample evidence – anecdotal as well as from surveys (e.g., <http://www.ambyssoft.com/surveys/>) – in the literature, to the best of our knowledge, there is very little theoretically grounded empirical research that affirms the benefits that agile methods provide vis-a-vis traditional approaches to software development. Our study fills this void by drawing on the extensive body of research on innovation diffusion. Following on the work by (Batra, VanderMeer & Dutta, 2011), our empirical study examines two dimensions of process agility: evolutionary development and process flexibility. The primary objective of this research is to examine how two seminal ideas in agile development – evolutionary development and process flexibility – affect the performance of individual developers in terms of productivity, quality of work, and the ease with which they are able to perform their jobs. Specifically, we focus on the mediating role of complexity and compatibility in affording benefits to software developers when they practice an approach characterized by evolutionary development and process flexibility. Based on (Siau, Long & Ling, 2010), the Unified Model of ISD Success provides a framework for studying specific factors that impact ISD success. Input consists of individual, team and organization factors (Siau, Long & Ling, 2010). Our study focuses on individual developers and examines specific process characteristics and how they interact to impact perceived relative advantage of the methodology. Although the decision to adopt a specific process is typically determined by the organization and not the individual developer, individual resistance can have a positive or negative impact on the benefits realized from its

28 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/an-empirical-investigation-of-the-perceived-benefits-of-agile-methodologies-using-an-innovation-theoretical-model/261028

Related Content

Controlled Experiments as Means to Teach Soft Skills in Software Engineering

Marco Kuhrmann, Henning Femmer and Jonas Eckhardt (2018). *Computer Systems and Software Engineering: Concepts, Methodologies, Tools, and Applications* (pp. 1355-1373).

www.irma-international.org/chapter/controlled-experiments-as-means-to-teach-soft-skills-in-software-engineering/192927

Software Module Clustering Using Bio-Inspired Algorithms

Kawal Jeet and Renu Dhir (2021). *Research Anthology on Recent Trends, Tools, and Implications of Computer Programming* (pp. 788-813).

www.irma-international.org/chapter/software-module-clustering-using-bio-inspired-algorithms/261054

Do Investments in ICT Help Economies Grow?: A Case of Transition Economies

Sergey Samoilenko (2019). *Handbook of Research on Technology Integration in the Global World* (pp. 40-63).

www.irma-international.org/chapter/do-investments-in-ict-help-economies-grow/208792

Trends in Peace Research: Can Cyber Détente Lead to Lasting Peace?

Nenad Putnik and Mladen Milošević (2018). *Cyber Security and Threats: Concepts, Methodologies, Tools, and Applications* (pp. 1694-1711).

www.irma-international.org/chapter/trends-in-peace-research/203581

Identification of Genomic Islands by Pattern Discovery

Nita Parekh (2012). *Computer Engineering: Concepts, Methodologies, Tools and Applications* (pp. 742-758).

www.irma-international.org/chapter/identification-genomic-islands-pattern-discovery/62476