

Chapter 99

Knowledge Management Initiatives in Agile Software Development: A Literature Review

Shanmuganathan Vasanthapriyan

 <https://orcid.org/0000-0002-0597-0263>

Sabaragamuwa University of Sri Lanka, Sri Lanka

ABSTRACT

Agile software development (ASD) is a knowledge-intensive and collaborative activity and thus Knowledge Management (KM) principals should be applied to improve the productivity of the whole ASD process from the beginning to the end of the phase. The goal is to map the evidence available on existing researches on KM initiatives in ASD in order to identify the state of the art in the area as well as the future research. Therefore, investigation of various aspects such as purposes, types of knowledge, technologies and research type are essential. The authors conducted a systematic review of literature published between 2010 and December 2017 and identified 12 studies that discuss agile requirements engineering. They formulated and applied specific inclusion and exclusion criteria in two distinct rounds to determine the most relevant studies for their research goal. Reuse of knowledge of the team is the perspective that has received more attention.

INTRODUCTION

Software development is a process which is a collection of steps like analyzing, designing, programming, documenting, testing, maintaining and bug fixing (Vasanthapriyan, Tian, & Xiang, 2015). It is clear that software development has organized to deliver final better-quality solutions to the end users or customers. When the term “software development” converts to the term “agile software development”, ASD has become a new turning point in the world. Although development teams have habituated traditional

DOI: 10.4018/978-1-6684-3702-5.ch099

software methods like waterfall and prototyping, it is a good forethought to use agile for the development teams worldwide (Dybå & Dingsøy, 2008).

There's a big focus in the ASD community on collaboration and the self-organizing team. That doesn't mean that there aren't managers. It means that teams have the ability to figure out how they're going to approach things on their own. It means that those teams are cross-functional. Those teams don't have to have specific roles involved so much as that when you get the team together, you make sure that you have all the right skill sets on the team.

Software development is a knowledge intensive and collaborative process which mainly depends on knowledge and experience of software engineers (Bjørnson & Dingsøy, 2008; Vasanthapriyan et al., 2015). Therefore, the knowledge of the members of a team and the outside team within the organization should be properly managed by capturing, storing and reusing when needed. Even though traditional software methods use detailed specifications and design upfront and rigorous documentations (Abrahamsen, Salo, Ronkainen, & Warsta, 2017) to manage the knowledge, agile methods and principals (Petersen, Feldt, Mujtaba, & Mattsson, 2008) emerge to the software development which brings collaboration and interaction within the team and the outside of the team. It helps to manage the knowledge more efficiently and effectively by presenting the right knowledge in the right form to the right person at the right time.'

KM is about making the right knowledge available to the right people. It is about making sure that an organization can learn, and that it will be able to retrieve and use its knowledge assets in current applications as they are needed. In the words of Peter Drucker it is "the coordination and exploitation of organizational knowledge resources, in order to create benefit and competitive advantage" (Drucker, 1999). Where the disagreement sometimes occurs is in conjunction with the creation of new knowledge. Wellman (2009) limits the scope of KM to lessons learned and the techniques employed for the management of what is already known. He argues that knowledge creation is often perceived as a separate discipline and generally falls under innovation management. Williams and Bukowitz (1999) link KM directly to tactical and strategic requirements. Its focus is on the use and enhancement of knowledge-based assets to enable the firm to respond to these issues. According to this view, the answer to the question "what is knowledge management" would be significantly broader.

A similarly broad definition is presented by Davenport and Prusak (1970), which states that KM "is managing the corporation's knowledge through a systematically and organizationally specified process for acquiring, organizing, sustaining, applying, sharing and renewing both the tacit and explicit knowledge of employees to enhance organizational performance and create value". The knowledge must be constructed in a social and evolutionary process involving all stakeholders during the software development. Nakamori introduced a generic knowledge construction framework "Theory of knowledge construction systems" which is a systems approach to synthesize a variety of knowledge and to justify new knowledge (Nakamori, 2013).

Alahyari reported that only a substantial amount of papers has been published in recent years topics related to agile software development. Most of them were related to particular agile methods or comparing agile and other development processes (Alahyari, Svensson, & Gorschek, 2017). However, no recent study was found to have a dedicated focus on the concept of knowledge management. Previous reviews and investigations have been conducted for KM in ASD in different perspectives such as principals, methods, pros and cons, opportunities (Neves, Rosa, Correia, & de Castro Neto, 2011) and supporting agile practices (Fowler & Highsmith, 2001). However, what are the influences of KM in ASD and what kind of tools help to proceed with KM in ASD is not well understood. To identify and address this gap, mapping study has been performed to determine that there is research evidence on a relevant topic. Re-

15 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/knowledge-management-initiatives-in-agile-software-development/294559

Related Content

A Novel Approach for Ontology-Based Feature Vector Generation for Web Text Document Classification

Mohamed K. Elhadad, Khaled M. Badranand Gouda I. Salama (2018). *International Journal of Software Innovation* (pp. 1-10).

www.irma-international.org/article/a-novel-approach-for-ontology-based-feature-vector-generation-for-web-text-document-classification/191205

Machine Learning Model to Predict Automated Testing Adoption

Muhammad Nouman Noor, Tamim Ahmed Khan, Farah Haneefand Muhammad Ismail Ramay (2022). *International Journal of Software Innovation* (pp. 1-15).

www.irma-international.org/article/machine-learning-model-to-predict-automated-testing-adoption/293268

Validating Security Design Pattern Applications by Testing Design Models

Takanori Kobashi, Nobukazu Yoshioka, Haruhiko Kaiya, Hironori Washizaki, Takano Okuboand Yoshiaki Fukazawa (2014). *International Journal of Secure Software Engineering* (pp. 1-30).

www.irma-international.org/article/validating-security-design-pattern-applications-by-testing-design-models/121680

A Method to Design a Software Process Architecture in a Multimodel Environment: An Overview

Mery Pesantes, Jorge Luis Risco Becerraand Cuauhtémoc Lemus (2014). *Agile Estimation Techniques and Innovative Approaches to Software Process Improvement* (pp. 219-242).

www.irma-international.org/chapter/a-method-to-design-a-software-process-architecture-in-a-multimodel-environment/100280

Speech Synthesis of Emotions Using Vowel Features

Kanu Boku, Taro Asada, Yasunari Yoshitomiand Masayoshi Tabuse (2013). *International Journal of Software Innovation* (pp. 54-67).

www.irma-international.org/article/speech-synthesis-emotions-using-vowel/77618