Chapter 115
A Tale of Two Agile Requirements Engineering Practices

Pankaj Kamthan
Concordia University, Canada

Terrill Fancott
Concordia University, Canada

ABSTRACT

There are a number of concerns in agile software development, including requirements engineering. There are different types of agile requirement, of which currently the most common forms are use cases and user stories. The use cases and user stories have different origins, both in space and in time, but by being among the practices of scenario-oriented requirements engineering (SORE), they are not entirely unrelated. The purpose of this chapter is to situate use cases and user stories in the context of each other. This is done by means of a conceptual framework for systematically comparing use cases and user stories. The understanding of similarities and differences between use cases and user stories have pedagogical as well as practical implications.

INTRODUCTION

The discipline of software engineering advocates a systematic and disciplined approach towards the development and evolution of software systems. The ecosystem of software engineering is changing constantly, and influences the practice of software engineering. In the past decade, there have been a number of notable strategic changes in industrial software engineering, including the adoption of agile methodologies (Highsmith, 2009). If industrial surveys are any sign, this trend is likely to continue for the foreseeable future.

There are a number of concerns in agile software development, including requirements engineering (Sillitti & Succi, 2005; Zhu, 2009; Leffingwell, 2011). There are different types of agile requirement, of which currently the most common forms are use cases (Jacobson et al., 1992) and user stories (Cohn, 2004). The use cases and user stories have different origins, both in space and in time, but, by being among the practices of scenario-oriented requirements engineering (SORE) (Alexander, 2011), they

are not entirely unrelated. The purpose of this article is to situate use cases and user stories in context of each other.

The need for a comparison among concepts arises naturally in many disciplines, and agile requirements engineering is no exception. Indeed, it is suggested by theories of learning, such as constructivism (Piaget, 1952), that upon initial exposure to a new concept C, a comparison between C and other, closely related and relatively more established, concepts C' is inevitable. This comparison is a necessary prerequisite for creating an understanding of C through assimilation and accommodation, selection, and application. In this article, C is user story and C' is use case.

The rest of the article is organized as follows. First, background and related work are presented. This is followed by introduction of a conceptual framework for systematically comparing use cases and user stories. Next, directions for future research are highlighted. Finally, concluding remarks are given.

BACKGROUND

In this section, a brief background on agile methodologies and requirements engineering is presented, and previous work on comparing use cases and user stories is analyzed.

Agile Methodologies

In the 1990s, a number of limitations of rigidity in approaches for development of certain types of software systems were realized. These limitations made it difficult for organizations that aimed for relatively short release cycles of their products to remain competitive in the market, had relatively small budgets to allocate for their projects, and required richer communication between technical and non-technical stakeholders (including customers and users) to address their needs better. The drive to cope with these limitations led to the inception of agility.

The *Agile Manifesto* constitutes the basis for a number of agile methodologies, including Agile Experience Design (AXD), Crystal Clear, Extreme Programming (XP), OpenUP, Scrum, and User-Centered Agile Process (UCAP). The agile methodologies have evolved over time and in certain cases, such as the Discipline Agile Delivery (DAD) process framework (Ambler & Lines, 2012), elements of multiple agile methodologies have been included.

Requirements Engineering

In software engineering, significant emphasis is placed on the requirements engineering, in general, and practices of requirements engineering, in particular, in the light of their impact on the later phases of software development and eventually their influence on the success of the software project (Wiegers, 2003). This causality is depicted in Figure 1. In requirements engineering, the attention is on the problem so as to devise a desirable and viable solution, namely the software system, which can satisfy the stakeholders.

The agile methodology underlying an agile project determines the type of agile requirement to be adopted. For example, in Crystal Clear and OpenUP, an agile requirement can take the form of a use case (Jacobson et al., 1992) and in AXD, XP, Scrum, and UCAP, an agile requirement can take the form of a user story (Cohn, 2004).

11 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/a-tale-of-two-agile-requirements-engineering-practices/214721

Related Content

MICA: A Mobile Support System for Warehouse Workers
Christian R. Prause, Marc Jentschand Markus Eisenhauer (2011). *International Journal of Handheld Computing Research (pp. 1-24).*
www.irma-international.org/article/mica-mobile-support-system-warehouse/51571

An Adaptive Backoff Algorithm for Mobile Ad-Hoc Networks
Yaser Khamayseh, Muneer Bani Yassein, Iman I. Badranand Wail Mardini (2011). *International Journal of Mobile Computing and Multimedia Communications (pp. 1-19).*
www.irma-international.org/article/adaptive-backoff-algorithm-mobile-hoc/55864

An Entrepreneurial Approach to Cloud Computing Design and Application: Technological Innovation and Information System Usage
Vanessa Ratten (2018). *Mobile Commerce: Concepts, Methodologies, Tools, and Applications (pp. 190-203).*
www.irma-international.org/chapter/an-entrepreneurial-approach-to-cloud-computing-design-and-application/183287

Viterbi Decoder in Hardware
Mário Pereira Véstias (2019). *Advanced Methodologies and Technologies in Network Architecture, Mobile Computing, and Data Analytics (pp. 1138-1152).*
www.irma-international.org/chapter/viterbi-decoder-in-hardware/214688

Mobile-Assisted Language Learning From Language Instructors' Perspectives
Brenda I. Lopez Ortizand Mara J. Luna (2019). *Advancing Mobile Learning in Contemporary Educational Spaces (pp. 60-87).*
www.irma-international.org/chapter/mobile-assisted-language-learning-from-language-instructors-perspectives/234048