Risk management analysis in Scrum software projects

Article *in* International Transactions in Operational Research · March 2017

DOI: 10.1111/itor.12401

3 authors:

Breno Gontijo Tavares
Instituto Nacional de Telecomunicações

Carlos E S Silva
Universidade Federal de Itajubá (UNIFEI)

Adler Diniz de Souza
Universidade Federal de Itajubá (UNIFEI)

INTERNATIONAL TRANSACTIONS IN OPERATIONAL RESEARCH

WILEY

INTERNATIONAL TRANSACTIONS IN OPERATIONAL RESEARCH

Intl. Trans. in Op. Res. 00 (2017) 1–22
DOI: 10.1111/itor.12401

Risk management analysis in Scrum software projects

Breno Gontijo Tavares[a], Carlos Eduardo Sanches da Silva[a] and Adler Diniz de Souza[b]

[a] *IEPG, Universidade Federal de Itajuba, Minas Gerais, Brazil*
[b] *IMC, Universidade Federal de Itajuba, Minas Gerais, Brazil*
E-mail: breno.tavares@inatel.br [Tavares]; sanches@unifei.edu.br [da Silva]; adler@unifei.edu.br [de Souza]

Received 10 March 2016; received in revised form 6 January 2017; accepted 22 January 2017

Abstract

One of the reasons for the failure of software projects is the absence of risk management procedures or its improper application. The adoption of Scrum in software projects is increasing. However, such approach does not specify risk management activities. This paper presents the results of a survey conducted using a qualitative approach to analyze how risk management is carried out in Scrum software projects. Consequently, we present risk management practices that achieved greater and lesser agreement among respondents and the literature, respectively. We found that risk management must be applied continuously in a feedback loop. Furthermore, Scrum projects must not have a high formal planning level, even for high-risk ones. The research verified that risk management in Scrum is performed differently from its application in traditional approaches. The framework has native resources, but classic processes of risk management would be incorporated and adapted.

Keywords: software engineering; Scrum; risk management

1. Introduction

Software projects are typically complex and they are particularly susceptible to failure (Bannerman, 2008). Most of these projects run over budget, and are terminated prematurely or fall far from meeting user expectations and business functionalities (Kaur and Sengupta, 2011). In this scenario, software development industries have recently been using agile methodologies to manage projects instead of traditional ones (Forrester Research, 2005; West and Grant, 2010), because they are generally considered heavy, unlike agile ones, which aim to provide light approaches to projects (Erickson et al., 2005). Furthermore, agile methodologies focus on managing and speeding up development activities (Goodpasture, 2010).

The Scrum framework is the most widely used among agile methodologies in software project management (Mahnic, 2010; Garzás and Paulk, 2013; Alharbi and Qureshi, 2014; VersionOne, 2016). It provides a set of good practices aimed at fast delivery value to the customers and can

Published by John Wiley & Sons Ltd, 9600 Garsington Road, Oxford OX4 2DQ, UK and 350 Main St, Malden, MA02148, USA.

simplify the work process, reduce development time, and enhance organizational transparency (Thamhain, 2013). However, risk management, which can reduce uncertainty and increase the chances of success in software projects (Charette, 2005; SEI, 2010; Chowdhury and Arefeen, 2011; Wet and Visser, 2013), is conducted mostly in an implicit way in projects that use agile methodologies (Nelson et al., 2008; Nyfjord and Kajko-Mattsson, 2008; Khatri et al., 2014; Moran, 2014). Nyfjord and Kajko-Mattsson (2008) have performed a comparative analysis between traditional and agile risk management approaches. The authors assert that agile approaches do not provide any risk management taxonomy, and they suggest integrating traditional practices to ensure an effective risk management.

Furthermore, studies regarding risk management applied to software development projects that implement agile methodologies are scarce (Hijazi et al., 2012; Tavares, 2015), and do not emphasize the process of how a team establishes, ranks, and takes action regarding risks (Smith and Pichler, 2005). The aim of this study is to understand how risk management is performed in projects that use the Scrum framework, with the following objectives:

- identify applied risk management practices in software development;
- conduct a case study in software projects that use Scrum to analyze the respondent's agreement with the risk management practices identified in scientific literature;
- analyze the agreement among the respondents and the standard respondent;
- rank the risk management practices according to the results.

This research is conducted for the following reasons:

- Theme relevance—There has been a growth in the number of software projects (Sethi et al., 2011; Bazaz et al., 2012); project risk in this segment (Bazaz et al., 2012); and in the use of Scrum (VersionOne, 2016), which is the most popular agile methodology (Azizyan et al., 2011; Meier and Ivarsson, 2013; Mundra et al., 2013; VersionOne, 2016).
- Use of risk management in software projects—Generally, it is observed that there exists a lack of risk management in software projects using Scrum (Hijazi et al., 2012; Tomanek and Juricek, 2015), and software projects in general (Bannerman, 2008; Junior et al., 2012; Singh et al., 2012).
- Software project success—Risk management can increase the chance of success in software projects (SEI, 2010; Chowdhury and Arefeen, 2011; Bannerman, 2015) by reducing their uncertainties and chances of failure (Wet and Visser, 2013). On the other hand, the absence of risk management procedures is one of the reasons for the failure of these projects (Charette, 2005; Dey et al., 2007; Lu et al., 2010).
- Characteristics of risk management—Agile methodologies do not explicitly suggest risk management processes (Nelson et al., 2008; Nyfjord and Kajko-Mattsson, 2008; Khatri et al., 2014; Moran, 2014).

The paper is organized as follows. Section 2 presents a literature review regarding the Scrum framework and risk management; Section 3 presents the research classification, case study design, and protocol; Section 4 presents the data collection and its results; and finally, Section 5 presents the discussions, conclusions, and suggestions for further research.

2. Risk management in Scrum

Scrum is defined by its creators (Schwaber and Sutherland, 2013) as a structural framework used to manage complex products that allow integration of many processes or techniques. According to Farlex (2016), a framework is a structure that supports or attaches other items. Some authors define it as a methodology and not a framework. For instance, Garzás and Paulk (2013) define Scrum as a project management methodology based on the use of an iterative and incremental life cycle model in software development. On the other hand, Schwaber (2004) states that Scrum is an agile process or a framework for agile project management. It is a process for project management, and certainly not a methodology; if this was the case, it would be considerably heavy (Schwaber, 2004). This research uses Scrum based on its framework definition.

The team roles, events, artifacts, and rules are Scrum components. The roles are divided into Product Owner, Development Team, and Scrum Master. Scrum teams are multifunctional, which are able to complete the work without depending on people outside the team. Another Scrum team feature is self-organization, which enables a team to define the best way to perform the work without being led by someone outside it.

Scrum prescribes five events, also known as ceremonies, all of which have a specified maximum duration, which cannot be reduced or increased (Schwaber and Sutherland, 2013). These events are designed to allow for transparency and inspection of projects. If any event is not performed, it will result in a reduction of transparency and loss of opportunity of inspection and project adaptation. Figure 1 presents the Scrum life cycle.

The Scrum life cycle is divided into several stages, usually two to four weeks long, through iterations called Sprints, which facilitates monitoring of the product being developed and identifies impediments. Some authors define an impediment as a project risk (Marçal et al., 2007; Menezes et al., 2013), while others state that there are differences between these two concepts. Tomanek and Juricek (2015) and Szalvay (2016) define impediment as anything that keeps a team from being productive.

Project risks can be defined as a set of factors or conditions that may pose a threat to project success (Wallace et al., 2004). It is important to quantify a risk, evaluate the probability of its occurrence, and its potential impact on the project (Huang and Han, 2008). According to Schwaber and Sutherland (2013), the Scrum framework uses an iterative and incremental approach to optimize predictability and risk control. The authors state that the use of Sprints also supports risk management, as it limits risk to one calendar month of cost (Schwaber and Sutherland, 2013).

However, other authors believe that Scrum and other agile methodologies, in general, do not suggest specific activities to support risk management (Nelson et al., 2008; Nyfjord and Kajko-Mattsson, 2008; Khatri et al., 2014; Moran, 2014), and this management in Scrum is not as good as in traditional methodologies (Ravi et al., 2012). According to Ravi et al. (2012), Scrum serves only for risk identification and it does not offer ways to analyze and manage them.

This study uses the impediment definition proposed by Jakobsen and Johnson (2008), as a problem that has occurred and is affecting progression of the project. When risk management is used in Scrum projects, it enables the prevention of impediments by implementing proactive actions to prevent project risks from becoming further impediments (Jakobsen and Johnson, 2008).

Despite the importance of risk management for the success of a project, only few scientific studies were identified in a survey conducted in the following databases: Web of Science, Scientific Electronic

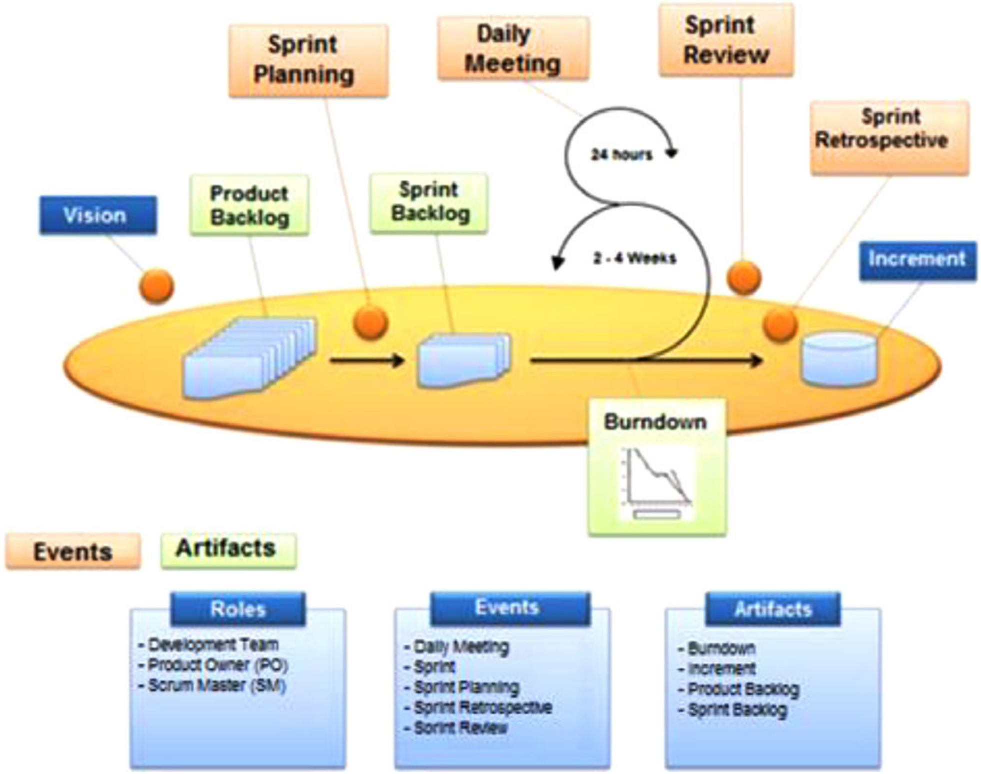


Fig. 1. The Scrum life cycle Adapted from Ettinger (2016). [Colour figure can be viewed at wileyonlinelibrary.com]

Library Online (SciELO), and Google Scholar. The terms “Scrum” and “Risk” and “Management” were searched in the article titles without returning any results.

3. Methodology

This study aims to contribute to the understanding of management practices in a real world context, where events cannot be controlled. Therefore, we have chosen a case study research methodology as proposed by Yin (2009).

For this study, we selected software projects that used Scrum. One of the selection criteria was the relevance of these projects in terms of invested effort with a minimum of 1500 hours for each project. As a result, 10 software projects that met the selection criteria were identified; among them, two were under development and eight were finished projects. Table 1 provides the project characteristics.

Five projects were developed for medium-sized companies while the other five were for large companies. This classification was based on the European Commission’s (2015) proposal, which establishes annual revenue at greater than 50 million euros for large companies, and between 10 and 50 million euros for medium-sized companies. We also used the *Banco Nacional de Desenvolvimento Econômico e Social* (BNDES) (2011) classification, which establishes annual revenue at greater than

Table 1
Summary of the characteristics of the 10 analyzed projects

Project	Status	Project duration	Worked hours	Team size	Company segment	Company size
A	Finished	Up to 1 year	5510.38	From 7 to 9 people	Industry	Large
B	Finished	Up to 1 year	6734.92	From 7 to 9 people	Industry	Large
C	In progress	Up to 1 year	1785.67	From 7 to 9 people	Industry	Large
D	Finished	From 1 to 3 years	16,766.78	From 7 to 9 people	Telecommunications	Medium
E	Finished	From 1 to 3 years	11,775.58	From 3 to 6 people	Telecommunications	Medium
F	Finished	Up to 1 year	6150.18	From 3 to 6 people	Telecommunications and Security Technology	Large
G	Finished	Up to 1 year	9090.59	From 3 to 6 people	Research and Development	Medium
H	Finished	Up to 1 year	7317.60	Greater than 9 people	Security Technology	Medium
I	Finished	Up to 1 year	1850.64	From 3 to 6 people	Security Technology	Large
J	In progress	Up to 1 year	3925.20	From 7 to 9 people	Telecommunications	Medium

300 million Brazilian reals for large companies, and annual revenue between 90 and 300 million Brazilian reals for medium-sized companies.

The projects were classified according to the diamond model, proposed by Shenhar and Dvir (2007). The model, also known as NTCP (Novelty, Technology, Complexity, and Pace), provides four dimensions for project classification. This model is considered to be a suitable framework for project classification (Silva and Jeronimo, 2013).

Scrum Masters of these projects responded to a structured questionnaire answered through an on-site interview. The NTCP classification was validated by senior managers of these projects through interviews. Table 2 presents the results of 10 projects according to the NTCP model.

According to Table 2, most projects were classified as "platform," "medium-tech," "system," and "fast/competitive." These classifications allow the identification of one of these research limitations, since applying the same research protocol in a different environment may present other results. Scrum Masters attended other on-site interviews to identify the implementation level of Scrum roles, events, artifacts, and rules in the 10 projects. This study was performed because of the possibility of not implementing Scrum entirely. According to Schwaber and Sutherland (2013), in the case of implementing only parts of Scrum and not the entire contents, the result will not be Scrum. However, in another publication (Sutherland and Schwaber, 2011), these authors believe that Scrum has a flexible structure that will be adapted to each context. For Salo and Abrahamsson (2008), the adaptation of Scrum is highly likely, as the application of this framework would differ from organization to organization and from project to project. The implementation of each characteristic of Scrum can be expensive and time consuming, which presents an obstacle to organizations (Machado et al., 2015). Other research adapted Scrum to be integrated the Capability Maturity Model Integration (CMMI) (Garzás and Paulk, 2013; Lukasiewicz and Miler, 2012). According to Beck (2000), the adaptation of Scrum to Extreme Programming (XP) and other agile practices is common.

Table 2
Result of projects classification according to NTCP model

NTCP		Projects A	B	C	D	E	F	G	H	I	J	Total (%)	
Novelty	Platform	•	•	•	•	•	•	•		•		8	80
	Derivative										•	1	10
	Breakthrough								•			1	10
Technology	Medium-tech		•	•			•	•			•	5	50
	High-tech	•			•	•			•			5	40
	Low-tech									•		1	10
	Super high-tech											0	0
Complexity	System	•	•	•	•	•	•		•		•	7	80
	Assembly							•		•		2	20
	Array											0	0
Pace	Fast/competitive	•	•	•		•	•	•	•	•	•	9	90
	Time critical				•							1	10
	Blitz											0	0
	Regular											0	0

In this study, Scrum is considered to be adaptable in effectively meeting the requirements of each project. Hence, the framework must not be used in a prescriptive way, but as a guide to good agile practices. The analysis was performed using four steps.

1. *Identifying Scrum practices*—This step aimed at lifting the main features of Scrum according to the *Scrum Guide* book (Schwaber and Sutherland, 2013). This guide contains the definitions of roles, events, artifacts, and rules of Scrum (Schwaber and Sutherland, 2013).
2. *Preparation of a questionnaire*—Seventeen closed questions, using a 6-point Likert scale, were classified according to the chapters of the *Scrum Guide*. Table 3 presents the questionnaire used.
3. *Performing interviews*—The Scrum Masters of the 10 projects were individually interviewed, using the structured questionnaire, and doubts were clarified using the *Scrum Guide* as reference. The interviews were conducted only for the Scrum Masters because of the knowledge that this role has on the framework and the project. According to Schwaber and Sutherland (2013), the Scrum Master is responsible for ensuring that Scrum is understood and applied, providing the team adherence to theory, practices, and rules. According to VersionOne (2014), approximately 44% of the 3501 respondents reported that Scrum Masters are those with the highest knowledge in agile methodologies.
4. *Analyzing the results*—Tables were constructed to facilitate the understanding of the project's scenario, providing the identification of smaller and larger adhesions of each project to Scrum.

Table 4 presents the results of the Scrum implementation level study. The results presented have not considered different weights to each of the questions because the reference used to create the questionnaire does not assign different weights to the Scrum features. The projects were ordered, as observed in Table 4, according to the median and variance level.

The differences in results between projects must be justified considering the different context of each project, where each one has defined a management scope to meet project-specific needs. This

Table 3
Questionnaire to identify the Scrum implementation level (complete table at https://goo.gl/j3bynV)

Scrum Practices		Questions		Classification
Nº	Description	Nº	Description	
1	Significant aspects of the process must be visible to those responsible for the outcome. Transparency requires those aspects be defined by a common standard so observers share a common understanding of what is being seen. For example: - A common language referring to the process must be shared by all participants; and, - Those performing the work and those accepting the work product must share a common definition of “Done” (Schwaber and Sutherland, 2013).	1	Significant aspects of the process are visible to those responsible for the outcome.	Scrum Theory
11	When a Product Backlog item or an Increment is described as “Done”, everyone must understand what “Done” means (Schwaber and Sutherland, 2013).	17	When a Product Backlog item or an Increment is described as “Done”, everyone must understand what “Done” means.	Artifact Transparency

setting allowed choosing the artifacts, events, and rules of Scrum that would be used in the project life cycle. To develop the research protocol, a survey was conducted in the three databases, which is presented in Table 5.

Table 6 presents the three search terms performed in the three databases. The search terms were combined using the Boolean “OR” operator, which entails that an article only had to include any one of the terms to be retrieved. That is, we searched 1 OR 2 OR 3, and identified 138 articles.

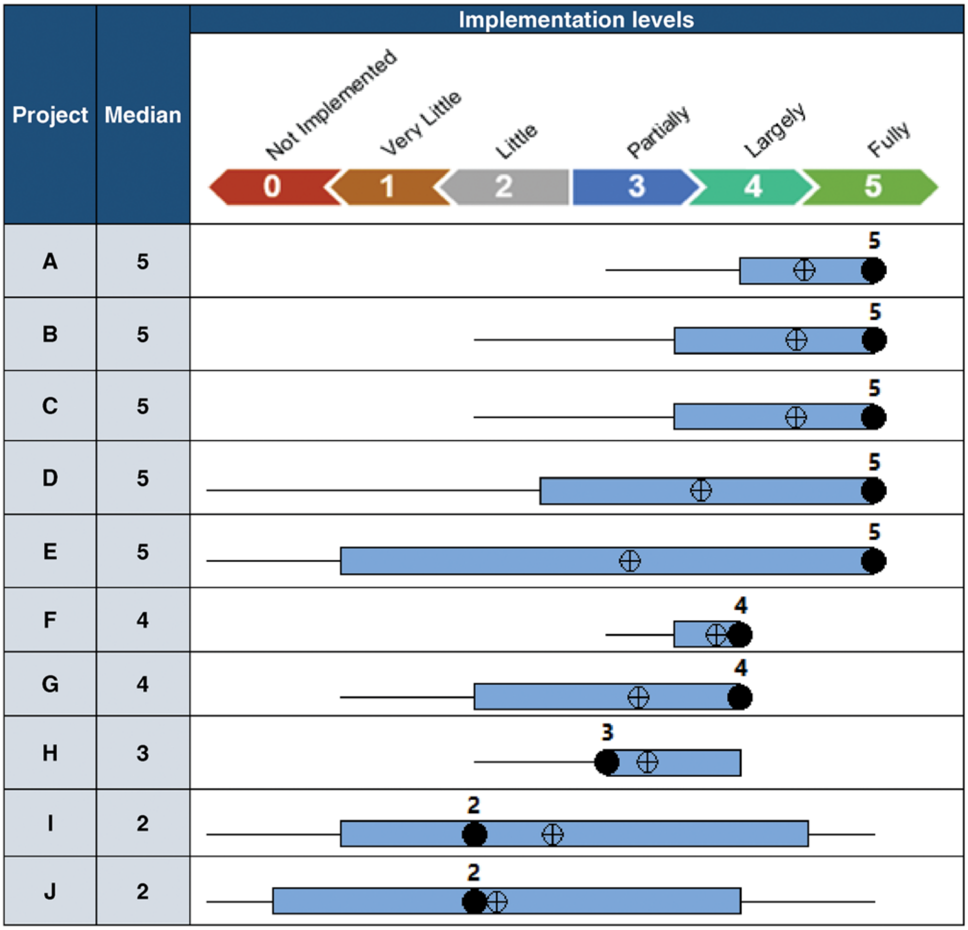
Another criterion for selection was defined aiming to improve the reliability of the research protocol. We made a survey on the impact factor (IF) of each article from the journals where they have been published.

Table 7 presents the total number of articles found in the databases, beyond those that possess IF from Journal Citation Reports (JCR). From the first 138 identified articles, only 11 possess IF from JCR. The research protocol development was based solely on these 11 articles.

They were analyzed aiming to gather the risk management practices. Three of the 11 analyzed articles did not contain any of those practices and thus could not be used in the research protocol development. The considered articles were not specifically regarding Scrum projects. However, they reference risk management applied to software projects. There was no article identified that specifically analyzes risk management of this framework in the selection criteria.

The eight articles were thoroughly analyzed and 35 risk management practices were identified considering only software projects. Sato and Hirao (2013), Sanches et al. (2009), and Hillson (2002) report the importance of performing the treatment of positive risks, also known as opportunities. Despite this, none of the identified practices specifically refers to the treatment of opportunities. However, it can be observed that 16 practices can be generally applied to risk management considering risks and opportunities. The identified practices were used to generate 40 questions, using a 6-point Likert scale, as it does not allow the respondents to choose a central point, which can be considered as neutral (Hair et al., 2005).

Table 4
Scrum implementation level



•, Median; ⊕, mean.

Table 5
Databases used to develop the research protocol

Database	Description
Web of Science	The Web of Science is the largest repository of scientific papers worldwide (Vanathi et al., 2015; Mukherjee et al., 2016)
SciELO	The SciELO database represents one of the major scholarly communication programs in the developing countries, and is a pioneer in the adoption of open access (Muñoz et al., 2013)
Google Scholar	Google Scholar is a free web-based database that indexes literature in a wide variety of formats (Swales and Leeder, 2012), and has steadily grown in importance in the academic library community (Cusker, 2013)

Table 6
Summary of the search performed in the databases

Search parameters		Number of articles found		
Filters used	Search terms	Web of Science	SciELO	Google Scholar
• The keywords were searched only in the article titles	(1) Software AND Project AND Risk AND Management	10	2	121
• The searches were restricted to the articles published between 2000 and 2015	(2) Agile AND Project AND Risk AND Management	1	0	4
	(3) Scrum AND Project AND Risk AND Management	0	0	0
• Web of Science database	Total		138 articles	

Table 7
Total articles found in the database and articles with IF from JCR

Database	Total no. of articles	Articles with IF
Web of Science	11	10
SciELO	2	0
Google Scholar	125	1
Total	138	11

Table 8
Pilot test application results

Total questions	Changed questions—1 pilot test	Changed questions—2 pilot test
40	23 57.50%	8 20%

The questionnaire was submitted to two pilot tests in two different companies that develop software projects and have used Scrum for more than four years. The objective of the pilot test was to ensure that the research instrument was well structured and allows the identification of protocol flaws that would be difficult to interpret by the respondents (Flynn et al., 1990). Table 8 presents the results for the application of the pilot tests.

From the application of the tests, 25 of the 40 questions were modified. After the first improvements, six of the reformulated questions on the first test received new improvement suggestions from the respondents of the second test. Considering the total sum of the modifications from both tests, there were 31 modifications. Table 9 presents the practices and their respective risk management questions, after the application of the pilot tests.

Beyond the research protocol of 40 questions, we developed 7 questions for external validation and 4 questions to supplement the questionnaire. According to Giuffre (1997a,b), external validation relates to the objective of inferring a probable situation based on the population as well as on the

Table 9
Risk management practices and questions to the research protocol (complete table at https://goo.gl/xOg8OB)

Nº	Description	Nº	Description
1	Senior management holds the key to establishing an organization that encourages 'functional' Risk Management behavior (Kwak and Stoddard, 2004).	1	Senior management holds the key to establishing an organization that encourages Risk Management behavior.
2	Project team members will be able to adapt Risk Management practices to their particular project environment if team members are well trained (Kwak and Stoddard, 2004).	2	Project team members will be able to adapt Risk Management practices to their particular project environment if team members are well trained.
3	As the size and complexity of the project increases, the effort for Risk Management increase exponentially (Kwak and Stoddard, 2004).	3	As the size and complexity of the project increases, the effort for Risk Management increase exponentially.
4	Real changes must occur in both management of the organization and behavior of individuals before Risk Management will improve (Kwak and Stoddard, 2004).	4	Real changes must occur in both management of the organization and behavior of individuals before Risk Management will improve.
32	Risk Management should be performed continuously in a feedback loop so that problematic situations can be dynamically detected and adjusted (Fan and Yu, 2004).	37	Risk Management should be performed continuously in a feedback loop so that problematic situations can be dynamically detected and adjusted.
33	We believe that the Risk Management of a project in terms of organizational perspectives requires a thorough knowledge of past experiences in previous projects (Esteves *et al.*, 2005).	38	The Risk Management requires a thorough knowledge of past experiences in previous projects.
34	We believe that other organizations can benefit from the creation of a database of risks for different projects and would help identify and manage project risks (Esteves *et al.*, 2005).	39	The creation of a database of risks for different projects help to identify and manage project risks.
35	Cannot be stated that formal Risk Management procedures produce better results than internal methods (ad hoc internally developed procedures) (Wet and Visser, 2013).	40	Cannot be stated that formal Risk Management procedures produce better results than internal methods (ad hoc internally developed procedures).

generalization of results. External validation is ensured by the reliability of the respondents. The seven applied questions aim to verify this reliability through the respondents' profile.

4. Discussion of results

The field research was performed throughout on-site interviews, providing a better communication between the interviewer and respondents. The interviews, ranging from 40 to 60 minutes, were conducted for 21 members of the selected projects. These were employees who had worked on the project and were still working in the organization. Table 10 presents the interviewees per project.

Scrum Masters of all the projects attended the interviews. On the other hand, not all the developers and Product Owners could attend because they were no longer working in the organization. Table 11 presents the results from the external validation. The interview identified that some of the professionals had certifications in the areas of project management, Scrum, and risk management (see Table 12). The interviewer has the three certifications represented on this table.

External validation was ensured through the profile of the interviewed professionals that work or worked in one of the 10 selected projects, besides being employees on specialist positions. This

Table 10
Interviewees per project

Project	Developers	Product Owners	Scrum Masters	Total
A	1	0	1	2
B	0	1	1	2
C	1	0	1	2
D	1	1	1	3
E	1	0	1	2
F	1	0	1	2
G	0	1	1	2
H	1	0	1	2
I	1	0	1	2
J	1	0	1	2
Total	8	3	10	10

Table 11
External validation results

Questions	Results
Education level	• 66.67% = Specialization • 33.33% = Graduation • 0% = Technician, high school, incomplete graduation, masters, and doctor's degree
Which functions have you performed?	• 90.48% = Developer • 47.62% = Scrum Master • 19.05% = Owner
How long is your experience in software projects development?	• 80.95% = Beyond 4 years • 9.52% = From 1 to 2 years • 9.52% = From 2 to 4 years • 0% = Less than 1 year
How long is your experience in Scrum?	• 42.86% = From 2 to 4 years • 33.33% = From 1 to 2 years • 14.29% = Less than 1 years • 9.52% = Beyond 4 years

Table 12
Professional certifications

Certification	No. of interviewed professionals
PMP—Project Management Professional (PMI, 2016aa)	4
PSM—Professional Scrum Master	2
PMI-RMP—Risk Management Professional (PMI, 2016bb)	1

information made it possible to understand the sample profile that assures the external validation of the questionnaire, making it necessary to verify the internal validation through Cronbach's alpha (Bryman and Bell, 2007), the theta (Θ) coefficient (Carmines and Zeller, 1979), and the omega (Ω) indicator (Heise and Bohrnstedt, 1970). They were calculated for the questionnaire with the aim of obtaining the internal consistency based on the average correlation between the items, and verify the coherence in the respondents' answers variation. The values were, 0.8302, 0.8320, and 0.8875, respectively, which are within the acceptable range (Heise and Bohrnstedt, 1970; Carmines and Zeller, 1979; Bryman and Bell, 2007).

An outlier analysis was performed to identify excessively distant data from the standard population (Osborne and Overbay, 2004; Hair et al., 2010; Pardoe, 2012). After its identification, the outliers were individually analyzed to verify if there were tangible explanations for excluding them. The questions referring external validation of the questionnaire were considered in this analysis with the intention of verifying if the respondent profile could justify the exclusion of any outlier. Despite this, all the outliers were retained after it was established that there were no identified nonstatistical explanations that reinforced the need for its exclusion. To obtain a deeper analysis for the answers, Table 13 was generated through Minitab 17®, providing a rank of risk management practices.

According to Table 13, the practice with the highest median and lowest variance is related to continuously accomplishing risk management in a feedback loop, where risk management must be updated at each iteration with new project data, and then generate new estimates (Fan and Yu, 2004). This implies that the respondents had higher agreement with this practice when it is regarding a Scrum project rather than other identified risk management practices. This agrees with the literature (Dingsøyr et al., 2006; Berczuk, 2007; Moe et al., 2010), which asserts that the Scrum life cycle possess ceremonies of which the focus is to obtain constant feedback. Dyba and Dingsøyr (2008) mention that continuous feedback is the key factor for project success.

The practice in the second place refers to the reuse of risk knowledge or the lessons learned. This result suggests that respondents believe this knowledge should be used in Scrum projects. The lessons learned are commonly used within Scrum life cycles and are mainly identified and discussed in retrospective meetings (Felker et al., 2012).

The third one is the practice that refers to the needs of changes in the organization management and individual behavior. The respondents generally agree with each other regarding this practice, justifying that individual behavior is a fundamental requirement for an effective achievement of risk management in Scrum projects. For Rosenberg (2015), the organizational culture and individual behavior interfere for the effective use of Scrum agile methodologies in general.

The practice in fourth place refers to the team's recognition that risk management is more than a methodological process. In general, the respondents agree with this practice once they believe that individual behavior and cooperation is important for Scrum project development. Maximini (2015) mentions that Scrum is more related to people, behavior, and culture than the processes.

On the other hand, this research identified that the risk management practice with the lowest result is related to the risk management tendency of being relegated to the Scrum Master after an initial round of risk identification. It shows that respondents in general agree less with this practice. In other words, they believe that risk management is also performed in other project events. Risk identification in Scrum is iterative, occurring during daily meetings (Marçal et al., 2007) or any other Scrum meeting (Ahola et al., 2014).

Table 13
Rank of risk management practices (complete table at https://goo.gl/sPZTxT)

	Practices		Questions	Median	Boxplot: Entirely Disagree 0 · Mostly Disagree 1 · Somewhat Disagree 2 · Somewhat Agree 3 · Mostly Agree 4 · Entirely Agree 5
32	Risk management should be performed continuously in a feedback loop so that problematic situations can be dynamically detected and adjusted (Fan and Yu, 2004).	37	Risk Management should be performed continuously in a feedback loop so that problematic situations can be dynamically detected and adjusted.	5	5
19	Since every project element may be related to uncertainty events that cause prejudice to a software project, development teams must reuse knowledge about risks that should occur in applications developed with these elements (Barros, Werner and Travassos, 2004).	21	The development teams must reuse knowledge about risks (lessons learned).	5	5
4	Real changes must occur in both management of the organization and behavior of individuals before risk management will improve (Kwak and Stoddard, 2004).	4	Real changes must occur in both management of the organization and behavior of individuals before Risk Management will improve.	5	5
28	Recognized that risk management is more than a methodological process (Bannerman, 2008).	30	Recognizing that Risk Management is more than a methodological process collaborates for an effective Risk Management.	5	5
20	By forcing risk documentation through scenarios, we expect that the risk identification team is able to precisely state their assumptions about the effects of these risks upon the project. Without formalizing the risk documentation, this reasoning may be highly subjective, eventually unusable to future projects (Barros, Werner and Travassos, 2004).	22	By forcing risk documentation through scenarios, the risk identification team is able to precisely state their assumptions about the effects of these risks upon the project. Without formalizing the risk documentation, this reasoning may be highly subjective, eventually unusable to future projects.	2	2
6	A project manager does not always have the time to implement a formal process into the system (Kwak and Stoddard, 2004).	6	A SCRUM MASTER does not always have the time to implement a formal process into the system.	2	2
16	Software development projects require high levels of Formal Planning for high Risk Exposure projects and low levels of Formal Planning for low Risk Exposure projects (Barki, Rivard and Talbot, 2001).	18	Software development projects require high levels of Formal Planning for high Risk Exposure projects and low levels of Formal Planning for low Risk Exposure projects.	2	2
30	After the initial round of risk identification, risk management tended to be relegated to the project manager, who often did little more than informally update the risk register before each steering committee meeting (Bannerman, 2008).	32	After the initial round of risk identification, Risk Management tended to be relegated to the SCRUM MASTER.	1	1

The practice with the second lowest result refers to high-risk projects that demand high formal planning levels and low-risk projects that demand less formal planning levels. The respondents, in general, did not agree with this practice, justifying that Scrum processes must be agile, even in high-risk projects, and a high-level formal planning impacts agility negatively. This opinion is different than that found in the literature, for example, Hoda et al. (2010) defend that agile methodologies must be used considering the project context for adapting the processes.

The practice with the third lowest result refers to the project manager's unavailability to implement a formal risk management process. This question is not regarding the project manager but the Scrum Master. This modification was performed to attend the specific needs of Scrum projects that do not possess the project manager position. The respondents, in general, believe that, in practice, the Scrum Master is not concerned with the implementation of formal processes, and that his or her role is rather focused on assisting the Development Team.

On the other hand, some authors claim that the Scrum Master is responsible for managing Scrum processes (Ktata and Lévesque, 2010) and other processes used for developing the software (Lárusdóttira et al., 2014). Qureshi and Albarqi (2015) claim that the Scrum Master is responsible for the implementation of the project risks plan. Therefore, this result does not agree with the Scrum literature.

The fourth lowest result relates to the practice that refers to the documentation of risks through scenarios, beyond the need of formalizing risk documentation. The respondents, in general, do not agree with this practice once they believe that the formalization for Scrum project must be cautious, evaluating first the specific needs for each project or even the staff experience on project development and risk treatment.

Some authors mention the importance of a formal documentation in Scrum projects. According to Costa et al. (2014), formal documentation is important to initiate the development phase, especially when the developed software is large and is being implemented in parallel by several teams. Read and Briggs (2012) mention that the lack of formal documentation of Scrum projects makes it harder for knowledge transfer from the project to big teams.

On the other hand, agile methodologies, including Scrum, discourage the use of formal documentation (Singh et al., 2014), but motivate a close cooperation and common and informal communication among the staff (Sharp et al., 2012). Machado et al. (2015) say that agile methodologies need less documentation to achieve high-quality software.

Aiming to analyze the agreement between practices, the kappa coefficient, an agreement measure used in the codification of textual information and calibration of the agreement level between two or more respondents, was calculated (Bryman and Bell, 2007). The calculation was performed based on the participants' answers and the standard answer for each question. The value varies between 0 and 1, being more reliable as it gets closer to 1. The coefficient was calculated on Minitab 17® using the agreement analysis tool, defined as the standard values of those established by the most experienced respondent, who has more than four years of experience developing software projects with Scrum, and experience on the three roles proposed by this framework and project management specialization. Table 14 presents the results of the agreement level analysis.

The practice related with the continuous accomplishment of risk management in a feedback loop, which had the highest result in median and variance analyses, also reached the highest agreement level, and was classified as "almost perfect" (Bryman and Bell, 2007). Another practice with one of the highest agreement levels refers to the need of change on the organization management and individual behavior, and was classified as "substantial" (Bryman and Bell, 2007). This practice is one of the first places in the rank. The practice referring to the need of the project risk management profile's adaption to its risk exposure level was also classified as "almost perfect." This result may indicate an agreement between the opinion from authors that define Scrum as a flexible and adaptable framework, making it possible to attend different needs and environments (Marchesi et al., 2007; Sutherland and Schwaber, 2011). The reuse of risk knowledge by the Development Teams was another practice classified as "almost perfect." The respondents and the standard respondent believe that the reuse on Scrum is vital once the framework is empiric and its application evolves according to the team's experience. This result is in agreement with existing literature as authors define Scrum as an empiric framework (Ravi et al., 2012; Meier and Ivarsson, 2013; Schwaber and Sutherland, 2013; Maximini, 2015).

Table 14
Agreement level rank (complete table at https://goo.gl/xnFuzb)

Nº	Practices		Questions	Values of Kappa	Interpretation of Kappa
32	Risk management should be performed continuously in a feedback loop so that problematic situations can be dynamically detected and adjusted (Fan and Yu, 2004).	37	Risk Management should be performed continuously in a feedback loop so that problematic situations can be dynamically detected and adjusted.	1,00	Almost perfect
4	Real changes must occur in both management of the organization and behavior of individuals before risk management will improve (Kwak and Stoddard, 2004).	4	Real changes must occur in both management of the organization and behavior of individuals before Risk Management will improve.	1,00	Almost perfect
19	Since every project element may be related to uncertainty events that cause prejudice to a software project, development teams must reuse knowledge about risks that should occur in applications developed with these elements (Barros, Werner and Travassos, 2004).	21	The development teams must reuse knowledge about risks (lessons learned).	1,00	Almost perfect
34	We believe that other organizations can benefit from the creation of a database of risks for different projects and would help identify and manage project risks (Esteves *et al.*, 2005).	39	The creation of a database of risks for different projects help to identify and manage project risks.	1,00	Almost perfect
[redacted]					
30	After the initial round of risk identification, risk management tended to be relegated to the project manager, who often did little more than informally update the risk register before each steering committee meeting (Bannerman, 2008).	32	After the initial round of risk identification, Risk Management tended to be relegated to the SCRUM MASTER.	0,45	Moderate
16	Software development projects require high levels of Formal Planning for high Risk Exposure projects and low levels of Formal Planning for low Risk Exposure projects (Barki, Rivard and Talbot, 2001).	18	Software development projects require high levels of Formal Planning for high Risk Exposure projects and low levels of Formal Planning for low Risk Exposure projects.	0,40	Moderate
31	Most commonly, one or more of four interrelated approaches to risk management are found in the literature and practice. These are checklists, analytical frameworks, process models, and risk response strategies (Bannerman, 2008).	34	The analytical frameworks are one of the most commonly interrelated approaches to Risk Management found in the literature and practice.	0,30	Fair
6	A project manager does not always have the time to implement a formal process into the system (Kwak and Stoddard, 2004).	6	A SCRUM MASTER does not always have the time to implement a formal process into the system.	0,25	Fair

On the other hand, for the practice with the lowest agreement level, project managers are unavailable to implement a formal process for risk management. This practice was also classified as one of the last in the ranking in Table 13. The respondents, in general, agree that during the project, Scrum Masters do not always have the time to implement such processes. The standard respondent, however, disagreed with this practice, justifying that the Scrum Master must always manage time to implement the necessary processes since it is one of the basic responsibilities of the role.

The practice with the second lowest agreement level refers to analytics frameworks being the most common risk management approach. The standard respondent agreed that such approaches are commonly found in literature and in Scrum projects. However, respondents generally disagree with this practice stating that analytical frameworks are rarely used in the literature and in practice. The respondents also state that, in Scrum projects, response strategies to risks and model processes are the most popular approaches. This agrees with Moran (2014) who does not mention analytical frameworks as one the tools used by agile methodologies for risk management.

The third lowest agreement level was the practice referring to high-risk projects demanding more formal planning levels and *vice versa*. The standard respondent agreed with this practice and claimed that Scrum processes must be adapted according to the needs of each project. This opinion agrees with Hoda et al. (2010) who defend that agile methodologies must be used considering the project context for adapting the processes. On the other hand, respondents, in general, believe that a high formal planning level should not be implemented by Scrum, because it impacts agility and does not respect The Agile Manifesto (2016) principles.

Finally, the practice with the fourth lowest agreement level refers to risk management tending to be relegated to the Scrum Master after the first round of risk identification. In general, the respondents did not agree that this behavior occurs in Scrum projects due to frequent meetings when using the framework. However, the standard respondent agreed that this behavior does occur

Table 15
Questions to supplement the questionnaire

Questions	Results
Does Scrum provide specific activities for the practice of risk management?	• 0% = Yes • 100% = No
Is risk management performed on Scrum similar to the one performed on traditional project management methodologies?	• 76.2% = No • 23.8% = Yes
What is the effectiveness of risk management in Scrum compared to the traditional methodologies of Project Management?	• 42.86% = It depends on the team experience • 33.33% = Better • 14.29% = Worse • 9.52% = Equal
How effective would the use of traditional methodologies of risk management in Scrum projects be?	• 38.10% = High effectiveness • 28.57% = Low effectiveness • 23.81% = It depends • 9.52% = Average effectiveness

in Scrum projects, depending on whether the Scrum Master is able to understand the importance of managing the project risks.

The agreement analysis also used respondents grouping by Scrum function and comparing it with the standard. The developers had the highest agreement level, followed by Scrum Masters and Owners, in that order. An analysis of variance (ANOVA) test was performed to analyze the variance of all the answers of the respondents. The results (p-value = 0.264; $\alpha = 0.05$) do not suggest statistically significant differences among the samples.

We also performed a cluster analysis, aimed at grouping and identifying important distribution and patterns at the data comprehension (Halkidi et al., 2001; Bagirov et al., 2003) and sample reduction (Manly, 2004). The respondents who had 80% or higher similarity result possess considerable disparities in their professional profiles, for example, projects, roles, and duration of experience with Scrum. Considering this scenario, the sample cannot be reduced.

Cluster analysis was also performed for the questions aiming to evaluate a possible questionnaire reduction. Fourteen questions were identified with an 80% or higher similarity, but only two of them possessed concepts related to each other. For this reason, they were reduced to one question. During the same interview, four additional questions were applied whose aim was to complement the risk management questionnaire in Scrum. Table 15 presents the issues and respective results.

According to Table 15, all the respondents believe that Scrum has no specific activities for the practice of risk management, but the ceremonies of the framework allow the treatment of risks, agreeing with the opinion of other authors (Nelson et al., 2008; Nyfjord and Kajko-Mattsson, 2008; Khatri et al., 2014; Moran, 2014).

For 76.2% of the respondents, risk management in Scrum is different from that practiced in traditional methodologies. The respondents mentioned that there are no defined processes for risk management in Scrum, while traditional methodologies have bureaucratic processes to implement this practice. Regarding the effectiveness of risk management, most respondents (48.86%) believe that it depends on the project team profile. The respondents justified that Scrum can bring better results in teams with significant experience in risk management. On the other hand, traditional methodologies will be more effective in less-experienced teams because they have specific activities of risk management, and present the necessary steps for its implementation. However, 33.33% of the respondents believe that risk management in Scrum is more effective and independent of a team's profile, because the risks are treated frequently during the framework ceremonies.

According to 38.10% of the respondents, the integration of traditional risk management practices to Scrum would be effective because they would improve this process in Scrum projects. On the other hand, 28.57% of the respondents believe that the effectiveness of this integration would be low, because the Scrum approach differs from that of traditional methods, which would leave the work heavy and slow rather than agile. In addition, 23.81% of the respondents believe that the effectiveness of this integration would depend on how it is performed. The adaptation of risk management is necessary to preserve principles presented by The Agile Manifesto (2016). Another justification of the respondents is that traditional risk management methodologies would bring greater efficiency to the Scrum depending on project risk, where higher risk projects would benefit from this integration. Some authors propose this integration in their research (Cobb, 2011; Conforto et al., 2014; Jahr, 2014).

5. Conclusions and future work

There has been an increase in the number of software projects adopting Scrum and it is becoming the most popular among the agile methodologies. Despite risk management being vital for project success, it is not widely used, and Scrum does not possess specific processes for its application.

We identified that the four most common risk management practices in Scrum projects are related to aspects of communication, lessons learned, and individuals' behavior. On the other hand, the four less common identified practices in Scrum refers to the Scrum Master responsibilities, implementation of formal planning, and use of risk management techniques and artifacts. Furthermore, Scrum projects should not have formal planning, and the Scrum Master should not focus on the implementation of formal processes.

The analyzed results indicate that risk management performed in Scrum projects is different from its application in traditional methodologies. The framework possesses native resources that contribute to risk identification and follow-up frequency, but the effectiveness of risk management in Scrum projects would depend on the team's experience.

Other classic risk management processes, such as planning, qualitative and quantitative analyses, and risk response plans, must be incorporated and adapted to Scrum. This adaptation would be performed preserving the benefits from this framework and The Agile Manifesto principles. As it is an empirical framework, an effective risk management application on Scrum relies on the project team's maturity. This scenario is empowered by the absence of some or many risk management formal processes.

In conclusion, software development companies can rely on the ranking of risk management practices presented by this study. They show the lack of risk management in Scrum and bring opportunities to integrate traditional risk management with this framework.

For the success of further research, it is suggested to carry out a grouping of the risk management practices according to artifacts, ceremonies, and roles on Scrum. The identification of the Scrum items that most contribute to risk management can also be carried out, perform the mapping, and highlight the point of attention on dealing with project risks. Subsequently, practices would be classified using the ORdinal CLASSification (ORCLASS) method (Larichev and Moshkovich, 1994) to determine the ones that should be implemented first. We also suggest to adapt classic risk management to the project needs and level of risk aversion, and integrate it with the projects.

Acknowledgments

The authors would like to express their acknowledgments to the three Brazilian research agencies, the CAPES Foundation, CNPq, and FAPEMIG, and especially all the interviewees and the reviewers.

References

Ahola, J., Frühwirth, C., Helenius, M., Kutvonen, L., Myllylahti, J., Nyberg, T., Pietikäinen, A., Pietikäinen, P., Röning, J., Ruohomaa, S., Särs, C., Siiskonen, T., Vähä-sipilä, A., Ylimannela, V., 2014. *Handbook of the Secure Agile Software Development Life Cycle*. University of Oulu, Oulu, Finland.

Alharbi, E., Qureshi, M.R.J., 2014. Implementation of risk management with SCRUM to achieve CMMI requirements. *International Journal. Computer Network and Information Security* 11, 20–25.

Azizyan, G., Magarian, M.K., Kajko-Mattson, M., 2011. Survey of Agile tool usage and needs. Agile Conference 2011, Salt Lake City, UT, pp. 29–38.

Bagirov, A.M., Rubinov, A.M., Soukhoroukova, N.V., 2003. An algorithm for clustering based on non-smooth optimization techniques. *International Transactions in Operational Research* 10, 611–617.

Banco Nacional de Desenvolvimento Econômico e Social (BNDES), 2011. National Bank for Economic and Social Development. Circular no. 34.

Bannerman, P.L., 2008. Risk and risk management in software projects: a reassessment. *Journal of Systems and Software* 81, 12, 2118–2133.

Bannerman, P.L., 2015. A reassessment of risk management in software projects. In Schwindt, C., Zimmermann, J. (eds) *Handbook on Project Management and Scheduling, Vol. 2*. Springer International, Cham, Switzerland. pp. 1119–1134.

Bazaz, Y., Gupta, S., Pakashrishi, O., Sharma, L., 2012. Comparative study of risk assessment models corresponding to risk elements. IEEE—International Conference on Advances in Engineering, Science and Management, ICAESM-2012, Tamil Nadu, India, pp. 61–66.

Berczuk, S., 2007. Back to basics: the role of Agile principles in success with a Distributed Scrum Team. Agile Conference, Washington, DC, pp. 382–388.

Bryman, A., Bell, E., 2007. *Business Research Methods* (2nd edn). Oxford University Press, New York.

Carmines, E.G., Zeller, R.A., 1979. *Reliability and Validity Assessment*. Sage, London.

Charette, R.N., 2005. Why software fails. *IEEE Spectrum* 42, 9, 42–49.

Chowdhury, A.A.M., Arefeen, S., 2011. Software risk management: importance and practices. *International Journal of Computer and Information Technology* 2, 1, 49–54.

Cobb, C.G., 2011. *Making sense of Agile project management: Balancing Control and Agility*. John Wiley & Sons, Hoboken, NJ.

Conforto, E.C., Salum, F., Amaral, D.C., Silva, S.L., Almeida, L.F.M., 2014. Can Agile project management be adopted by industries other than software development? *Project Management Journal* 45, 3, 21–34.

Costa, N., Santos, N., Ferreira, N., Machado, R.J., 2014. Delivering user stories for implementing logical software architectures by Multiple Scrum Teams. *Computational Science and Its Applications—ICCSA 8581*, 747–762.

Cusker, J., 2013. Elsevier Compendex and Google Scholar: a quantitative comparison of two resources for engineering research and an update to prior comparisons. *Journal of Academic Librarianship* 39, 3, 241–243.

de Wet, B., Visser, J.K., 2013. An evaluation of software project risk management in South Africa. *South African Journal of Industrial Engineering* 24, 1, 14–28.

Dey, P.K., Kinch, J., Ogunlana, S.O., 2007. Managing risk in software development projects: a case study. *Industrial Management and Data Systems* 107, 2, 284–303.

Dingsøyr, T., Hanssen, G.K., Dybå, T., 2006. Developing software with Scrum in a small cross-organizational project. Proceedings of the 13th European Conference on Software Process Improvement, Joensuu, Finland, pp. 11–13.

Dyba, T., Dingsøyr, T., 2008. Empirical studies of Agile software development: a systematic review. *Information and Software Technology* 50, 9–10, 833–859.

Erickson, J., Lyytinen, K., Siau, K., 2005. Agile modeling, Agile software develpment, and extreme programming: the state of research. *Journal of Database Management* 16, 4, 88–100.

Ettinger, D., 2016. A engrenagem do Scrum. Available at http://danielettinger.com/2011/04/06/a-engrenagem-do-scrum (accessed January 20, 2017).

European Commission, 2015. *User Guide to the SME Definition*. Internal Market, Industry, Entrepreneurship and SMEs, European Commission, Brussels.

Fan, C., Yu, Y., 2004. BBN-based software project risk management. *Journal of Systems and Software* 73, 193–203.

Farlex, 2016. The free dictionary. Available at http://www.thefreedictionary.com/framework (accessed January 30, 2017).

Felker, C., Slamova, R., Davis, J., 2012. Integrating UX with Scrum in an undergraduate software development project. SIGCSE'12 Proceedings of the 43rd ACM Technical Symposium on Computer Science Education, Grinnell, IA, pp. 301–306.

Flynn, B.B., Sakakibara, S., Schroeder, R.G., Bates, K.A., Flynn, E.J., 1990. Empirical research methods in operations management. *Journal of Operations Management* 9, 2, 250–284.

Forrester Research, 2005. *Software and Services in Large Enterprises, Business Technographics*. Forrester Research, Cambridge, MA.

Garzás, J., Paulk, M.C., 2013. A case study of software process improvement with CMMI-DEV and Scrum in Spanish companies. *Journal of Software: Evolution and Process* 25, 12, 1325–1333.

Giuffre, M., 1997a. Designing research survey design part one. *Journal of PeriAnesthesia Nursing* 12, 4, 275–280.

Giuffre, M., 1997b. Designing research survey design part two. *Journal of PeriAnesthesia Nursing* 12, 5, 358–362.

Goodpasture, J.C., 2010. *Project Management the Agile Way: Making it Work in the Enterprise*. J. Ross Publishing, Fort Lauderdale, FL.

Hair, J., Black, W., Babin, B., Anderson, R., 2010. *Multivariate Data Analysis* (7th edn). Prentice-Hall, Upper Saddle River, NJ.

Hair, J.F. Jr., Babin, B., Money, A.H., Samouel, P., 2005. *Fundamentos de métodos de pesquisa em administração*. Bookman, Porto Alegre.

Halkidi, M., Batistakis, Y., Vazirgiannis, M., 2001. On clustering validation techniques, *Journal of Intelligent Information Systems* 17, 2, 107–145.

Heise, D.R., Bohrnstedt, G.W., 1970. Validity, invalidity and reliability. In Borgatta, E.F., Bohrnstedt, G. (eds) *Sociological Methodology*, Jossey Bass, San Francisco, CA, pp. 104–129.

Hijazi, H., Khdour, T., Alarabeyyat, A., 2012. A review of risk management in different software development methodologies. *International Journal of Computer Applications* 45, 7, 8–12.

Hillson, D., 2002. Extending the risk process to manage opportunities. *International Journal of Project Management* 20, 3, 235–240.

Hoda, R., Kruchten, P., Noble, J., Marshall, S., 2010. Agility in context. Proceedings of the ACM International Conference on Object Oriented Programming Systems, Languages, and Applications, OOPSLA, Reno, NV, pp. 74–88.

Huang, S.-J., Han, W.-M., 2008. Exploring the relationship between software project duration and risk exposure: a cluster analysis. *Information and Management* 45, 3, 175–182.

Jahr, M., 2014. A hybrid approach to quantitative software project scheduling within Agile frameworks. *Project Management Journal* 45, 3, 35–45.

Jakobsen, C.R., Johnson, K.A., 2008. Mature Agile with a twist of CMMI. Agile '08 Conference, Toronto, Canada.

Junior, I.H.F., Azevedo, R.R., Moura, H.P., Silva, D.S.M., 2012. *Elicitation of communication inherent risks in distributed software development*. IEEE Seventh International Conference on Global Software Engineering Workshops, Porto Alegre, Brazil, pp. 37–42.

Kaur, R., Sengupta, J., 2011. Software process models and analysis on failure of software development projects. *International Journal of Scientific and Engineering Research* 2, 2, 1–4.

Khatri, S.K., Bahri, K., Johri, P., 2014. Best practices for managing risk in adaptive Agile process. International Conference on Reliability, Infocom Technologies and Optimization, ICRITO, Noida, India, pp. 1–5.

Ktata, O., Lévesque, G., 2010. Designing and implementing a measurement program for Scrum teams: what do Agile developers really need and want? Third Conference on Computer Science and Software Engineering, C3S2E, Montreal, QC, Canada. Association for Computing Machinery, pp. 101–107.

Kuhl, J.G., 2014. Incorporation of Agile development methodology into a Capstone Software Engineering Project Course. Proceedings, the 2014 ASEE North Midwest Section Conference, Iowa City, IA.

Larichev, O.I., Moshkovich, H.M., 1994. An approach to ordinal classification problems. *International Transactions in Operational Research* 1, 3, 375–385.

Lárusdóttira, M., Cajanderb, Å., Gulliksenc, J., 2014. Informal feedback rather than performance measurements—user-centred evaluation in Scrum projects. *Behaviour and Information Technology* 33, 11, 1118–1135.

Lu, W., Liang, C., Ding, Y., 2010. A method for risk assessment in IT project with incomplete information. International Conference on Management and Service Science, MASS, Wuhan, China, pp. 1–4.

Machado, T.C.S., Pinheiro, P.R., Tamanini, I., 2015. Project management aided by verbal decision analysis approaches: a case study for the selection of the best SCRUM practices. *International Transactions in Operational Research* 22, 287–312.

Mahnic, V., 2010. Teaching Scrum through team-project work: students' perceptions and teacher's observations. *International Journal of Engineering Education* 26, 1, 96–110.

Manly, B.F.J., 2004. *Multivariate Statistical Methods: A Primer* (3rd edn). Chapman and Hall/CRC Press, Boca Raton, FL.

Marçal, A.S.C., Freitas, B.C.C., Soares, F.S.F., Belchior, A.D., 2007. *Mapping CMMI Project Management Process Areas to SCRUM Practices*. Software Engineering Workshop, Columbia, MD, pp. 13–22.

Marchesi, M., Mannaro, K., Uras, S., Locci, M., 2007. Distributed Scrum in research project management. In Concas, G., Damiani, E., Scotto, M., Succi, G. (eds) *Agile Processes in Software Engineering and Extreme Programming, Lecture Notes in Computer Science* 4536, Springer Verlag, Berlin, pp. 240–244.

Maximini, D., 2015. *The Scrum Culture Introducing Agile Methods in Organizations*. Springer International Publishing, New York.

Meier, A., Ivarsson, J.C., 2013. Agile software development and service science. *Journal on Computing* 3, 3, 103–107.

Menezes JR, J., Gusmão, C.M.G., Moura, H., 2013. Defining indicators for risk assessment in software development projects. *CLEI Electronic Journal* 16, 1.

Moe, N.B., Dingsøyr, T., Dybå, T., 2010. A teamwork model for understanding an Agile team: a case study of a Scrum project. *Information and Software Technology* 52, 480–491.

Moran, A., 2014. *Agile Risk Management*. Springer, Heidelberg, pp. 33–60.

Mukherjee, S., Uzzi, B., Jones, B., Stringer, M., 2016. A new method for identifying recombinations of existing knowledge associated with high-impact innovation. *Journal of Product Innovation Management* 33, 2, 224–236.

Mundra, A., Misra, S., Dhawale, C.A., 2013. Practical Scrum-Scrum Team: way to produce successful and quality software. International Conference on Computational Science and Its Applications, ICCSA, Ho Chi Minh City, Vietnam, pp. 119–123.

Muñoz, P., Ontiveros, M., De La Barquera, S., Packer, A.L., 2013. Action lines for the years 2014–2016 with the objective of increasing the visibility of the SciELO network journals and collections. Reunião sobre linhas de ação para o desenvolvimento dos periódicos indexados no SciELO, Santiago de Chile, pp. 25–27.

Nelson, C.R., Taran, G., Hinojosa, L.L., 2008. Explicit risk management in Agile process. XP 2008, Limerick, Republic of Ireland, pp. 190–201.

Nyfjord, J., Kajko-Mattsson, M., 2008. Outlining a model integrating risk management and Agile software development. Proceedings of the 34th EUROMICRO Conference on Software Engineering and Advance Applications, IEEE Computer Society, Parma, Italy, pp. 476–483.

Osborne, J.W., Overbay, A., 2004. The power of outliers (and why researchers should always check for them). *Practical Assessment, Research and Evaluation* 9, 6.

Pardoe, I., 2012. *Applied Regression Modeling: A Business Approach*. Wiley, Hoboken, NJ.

PMI—Project Management Institute, 2016a. Project management professional (PMP). Available at http://www.pmi.org/certification/project-management-professional-pmp.aspx (accessed March 20, 2016).

PMI—Project Management Institute, 2016b. PMI risk management professional (PMI-RMP). Available at http://www.pmi.org/certification/risk-management-professional-rmp.aspx (accessed March 20, 2016).

Qureshi, M.R.J., Albarqi, A., 2015. Proposal of new PRORISK model for GSD projects. *International Journal of Information Technology and Computer Science* 6, 38–44.

Ravi, S.P., Reddaiah, B., Movva, L.S., Kilaparthi, R., 2012. A critical review and empirical study on success of risk management activity with respect to Scrum. *Engineering Science and Technology: An International Journal* 2, 3, 467–473.

Read, A., Briggs, R.O., 2012. The many lives of an Agile story: design processes, design products, and understandings in a large-scale Agile development project. 45th Hawaii International Conference on System Science, HICSS, Maui, HI, pp. 5319–5328.

Rosenberg, S., 2015. Organizational culture aspects of an Agile transformation. Agile processes in software engineering and extreme programming. 16th International Conference, XP 2015, Helsinki, Finland, May 25–29, pp. 279–286.

Sanches, H., Robert, B., Bourgault, M., Pellerin, R., 2009. Risk management applied to projects, programs, and portfolios. *International Journal of Managing Projects in Business* 2, 1, 14–35.

Salo, O., Abrahamsson, P., 2008. Agile methods in European embedded software development organisations: a survey on the actual use and usefulness of Extreme Programming and Scrum. *IET Software* 2, 1, 58–64.

Sato, T., Hirao, M., 2013. Optimum budget allocation method for projects with critical risks. *International Journal of Project Management* 31, 1, 126–135.

Schwaber, K., 2004. *Agile Project Management with Scrum*. Microsoft Press, Redmond, WA.

Schwaber, K., Sutherland, J., 2013. *The Scrum Guide*. Scrum.org.

SEI—Software Engineering Institute, 2010. *CMMI—Capability Maturity Model Integration*, Version 1.3. Carnegie Mellon University, Pittsburgh, PA.

Sethi, T.S., Hari, C.V.M.K., Kaushal, B.S.S., Sharma, A., 2011. Cluster analysis and Pso for software cost estimation. *Information Technology and Mobile Communication* 147, 281–286.

Sharp, H., Giuffrida, R., Melnik, G., 2012. Information flow within a dispersed Agile team: a distributed cognition perspective. In Wohlin, C. (ed.) *Agile Processes in Software Engineering and Extreme Programming, Lecture Notes in Business Information Processing*, Vol. 111. Springer, Berlin, Heidelberg, pp. 62–76.

Shenhar, A.J., Dvir, D., 2007. *Reinventing Project Management: The Diamond Approach to Successful Growth and Innovation*. Harvard Business School Press, Boston, MA.

Silva, M., Jeronimo, C., 2013. From Taylor to tailoring—in pursuit of the organizational fit. Second International Scientific Conference on Project Management in the Baltic Countries, Riga, Latvia.

Singh, A., Singh, K., Sharma, N., 2014. Agile knowledge management: a survey of Indian perceptions. *Innovations in Systems and Software Engineering* 10, 4, 297–315.

Singh, B., Sharma, K.D., Chandra, S., 2012. A new model for software risk management. *International Journal of Computer Technology and Applications* 3, 3, 953–956.

Smith, P.G., Pichler, R., 2005. Agile risks, Agile rewards. *Software Development Magazine* 13, 50–53.

Sutherland, J., Schwaber, K., 2011. *The Scrum Papers: Nut, Bolts, and Origins of an Agile Framework*. Scrum, Inc., Cambridge, MA.

Swales, J.M., Leeder, C., 2012. A reception study of the articles published in *English for Specific Purposes* from 1990–1999. *English for Specific Purposes* 31, 137–146.

Szalvay, V., 2016. *Glossary of Scrum Terms*. Available at https://www.scrumalliance.org/community/articles/2007/march/glossary-of-scrum-terms#1126 (accessed February 9, 2016).

Tavares, B.T., 2015. Risk management analysis in software projects that use the Scrum framework. Masters dissertation, Federal University of Itajubá, Itajubá.

Thamhain, H., 2013. Managing risks in complex projects. *Project Management Journal* 44, 2, 20–35.

The Agile Manifesto, 2016. Principles behind the Agile manifesto. Available at http://www.agilemanifesto.org/principles.html (accessed January 21, 2017).

Tomanek, M., Juricek, J., 2015. Project risk management model based on PRINCE2 and Scrum frameworks. *International Journal of Software Engineering and Applications*, 6, 1, 81–88.

Vanathi, B., Saravanan, T., Nagarajan, M., 2015. Growth of literature in chemistry research output in Tamil Nadu universities: a scientometric study, 1989–2015. *Journal of Advances in Library and Information Science* 43, 3, 187–190.

VersionOne, 2014. *8th Annual State of Agile Survey*. VersionOne, Inc., Atlanta, GA.

VersionOne, 2016. *10th Annual State of Agile Survey*. VersionOne, Inc., Atlanta, GA.

Wallace, L., Keil, M., Rai, A., 2004. How software project risks affect project performance: an investigation of the dimensions of risk and an exploratory model. *Decision Sciences* 35, 2, 289–321.

West, D., Grant, T., 2010. *Agile Development: Mainstream Adoption Has Changed Agility*. Forrester Research, Cambridge, MA.

Yin, R.K., 2009. *Case Study—Planning and Method* (2nd edn). Bookman, São Paulo.