Recibido 1 Feb 2018 *ReCIBE, Año 7 No. 1, Mayo 2018*
Aceptado 2 Abr 2018

Factors Models of Scrum Adoption in the Software Development Process: A Systematic Literature Review

Modelos de Factores en la Adopción de Scrum in el Proceso de Desarrollo de Software. Una Revisión Sistemática de la Literatura

Marilyn Sihuay
msihuayr@unmsm.edu.pe
Escuela de Postgrado, Universidad Nacional Mayor de San Marcos, Lima, Perú

Abraham Dávila
abraham.davila@pucp.edu.pe
Departamento de Ingeniería, Pontificia Universidad Católica del Perú, Lima, Perú

Marcelo Pessoa
mpessoa@usp.br
Polytechnic School, University of Sao Paulo, Sao Paulo, Brazil

Abstract: (Background) The adoption of Agile Software Development (ASD), in particular Scrum, has grown significantly since its introduction in 2001. However, in Lima, many ASDs implementations have been not suitable (uncompleted or inconsistent), thus losing benefits obtainable by this approach and the critical success factors in this context are unknown. (Objective) To analyze factors models used in the evaluation of the adoption of ASDs, as these factors models can contribute to explaining the success or failure of these adoptions. (Method) In this study we used a systematic literature review. (Result) Ten models have been identified; their similarities and differences are presented. (Conclusion) Each model identified consider different factors, however some of them are shared by five of these models, such as team member attributes, engaging customer, customer collaboration, experience and work environment.

Keywords: Software Process Model, Process Adoption, Agile Software Development, Systematic Literature Review.

Resumen: (Antecedentes).La adopción del Desarrollo de Software Agile (DSA), en particular Scrum, ha crecido significativamente desde su introducción en 2001. Sin embargo, en Lima, muchas implementaciones de DSA no han sido adecuadas (incompletas o inconsistentes), perdiendo así los beneficios que se pueden obtener con este enfoque y los factores de éxito críticos en este contexto son desconocidos. (Objetivo) Analizar los modelos de factores utilizados en la evaluación de la adopción de ASD, ya que estos modelos de factores pueden contribuir a explicar el éxito o el fracaso de estas adopciones. (Método) En este estudio, utilizamos una revisión sistemática de la literatura. (Resultado) Diez modelos han sido identificados; sus similitudes y diferencias son presentadas. (Conclusión) Cada modelo identificado considera diferentes factores, sin embargo, algunos de ellos son compartidos por cinco de estos modelos, tales como los atributos del miembro del equipo, el compromiso del cliente, la colaboración del cliente, la experiencia y el entorno laboral.

Palabras Clave: Modelo de Proceso Software, Adopción de Proceso, Desarrollo de Software Agile, Revisión Sistemática de la Literatura.

1. Introduction

The software industry has changed significantly in the way software is developed. The software process has evolved from models with long phases and intermediate results towards models with frequent delivery aimed at satisfying customer needs (Azevedo Santos, 2011) (Dingsøyr, Nerur, Balijepally, & Brede Moe, 2012). With the publication of the Agile Manifesto in 2001, a collection of principles were expressed in order to address the change leaving traditional methods (Agile Manifesto, 2001). These principles emphasized interactions and conceptual simplicity, development-oriented and fast delivery, intense customer collaboration, high quality, low costs and dynamism to face constant changes in the project (Agile Manifesto, 2001). According to Agile Alliance, Agile Software Development (ASD) covers methods and practices based on Agile Manifesto (Agile Alliance, 2016). In the ASD context, the two most important aspects are productivity and quality (Kumar & Kumar Bhatia, 2012). Moreover, according to some authors, traditional software development methods do not conform to the current trends of businesses and technologies, where changes are frequent, affecting the organization and work team, decision-making, management requirements, relationships with partners or suppliers and organizational culture (Hass, 2009), (Chan & Thong, 2007), (Stankovic, Nikolic, Djordjevic, & Cao, 2013), (Chow & Cao, 2008), (Dyba, 2000), (Imreh & Raisinghani, 2011).

Other authors note that ASD are intended to support software development in organizations which aims to introduce and expand their products and services in dynamic markets (Cao, Mohan, Xu, & Ramesh, 2009), (Highsmith & Cockburn, 2001), (Bohem, 2002). The ASD practices have been well received in the industry, favoring its adoption in a growing number of companies (Zhang, Hu, Dai, & Li, 2010). Additionally, agile principles have been introduced in other domains such as project management (Mark, 2011).

Despite the useful principles of ASD, their adoption is a process that involves significant challenges (Pagrut, 2008), (Oyeyipo, 2011). According to the authors experience and preliminary literature review, this adoption also generates problems (Pagrut, 2008), (Dubakow, 2010). Since ASD are also a way to address processes, it is sensitive to people, involved technologies or the market itself; even at the beginning of a process of adoption (Agile Manifesto, 2001). According to Dubakow (Dubakow, 2010), (Mishra & Mishra, 2011), (Hajjdiab & Taleb, 2011), (Kanane, 2014), some problems during the adoption of ASD are: (1) to start using tools or processes before getting familiar with the method; (2) to use it only in development activities; (3) to use it without considering techniques; due to its importance for a balance between architecture and communication; (4) to consider the Scrum master as the project manager who assigns the tasks; (5) to locate team members base on the roles in the project; (6) the coach is not the right person to handle the adoption; (7) to not gather requirements from the

customer resulting in product redefinition; (8) the lack of a self-organized teams; which require a leader to guide the team to a clear objective; overzealous teams do not have the sufficient experience to implement agile methods; (9) to fail in managing sprint issues such as duration, work load, changes and freezing conditions of the sprints; (10) to perform testing on each sprint without considering non-functional requirements verifications; (11) to skip daily meetings because of current work pressure; and (12)) to maintain a traditional culture, bureaucratic structure and old documentation habits.

Scrum adoption must begin with a cultural change of the people involved and its success depends on the hard work and passion of the individuals (Dubakow, 2010). Other reported problems such as lack of information, lack of appropriate architecture planning, low test coverage (Cao, Mohan, Xu, & Ramesh, 2009) and limited knowledge of Scrum (Abrahamsson, Salo, Ronkainen, & Warsta, 2002), (Conboy & Fitzgerald, 2010); and among others are the main inhibitors to conduct an organizational innovations (Daghfous & White, 1994).

In our experience, we can say that Scrum is not successfully implemented due to an inadequate control where there is not a recognized Scrum Master in the organization. There are only some aspects of the framework that are actually implemented such as the daily Scrum meetings, retrospective meetings and the use of user stories, but these are disproportioned among the meetings of the project teams.

The adoption of a process model and its subsequent use are subject to a number of factors that influence the obtained results. As per Rogers, there are five factors in the adoption of an innovation (Rogers, 2003): (i) perceived relative advantage among users, (ii) compatibility with their needs and expectations, (iii) simplicity to be understood and implemented, (iv) initial trialability, and (v) observable benefits. Also, Fichman and Kemerer consider that there are limits in the adoption of an innovation such as the lack of diffusion of the innovation regarding their advantages, the complexity and compatibility, in additions to the lack of knowledge of the innovation's application (Fichman & Kemerer, 1999). Other relevant factors perceived in the context of Scrum are: lack of commitment of the management to engage managers (Dyba, 2005), (Goodman, 1996), (Kasse & McQuaid, 2000), (Niazi, Wilson, & Zowghi, 2006), (Powell, 1995), lack of clear objectives (Dyba, 2005), (Kasse & McQuaid, 2000), (El Emam, Goldenson, McCurley, & Herbsleb, 1998) especially inexperienced technical teams (Baddoo & Hall, 2003), (Goodman, 1996), (Kasse & McQuaid, 2000), (Niazi, Wilson, & Zowghi, 2006), lack of training to learn new methods and techniques (Niazi, Wilson, & Zowghi, 2006), (Powell, 1995), pilot syndrome which limits the expansion of benefits just to the pilot project and not the whole company (Goodman, 1996), (Repenning & Sterman, 2001), lack of measurement to demonstrate the success of the adoption (Dyba, 2000), (Niazi, Wilson, & Zowghi,

2006), (Powell, 1995), (Rousseau & McCarthy, 2007), process-oriented rather than results-oriented which include the approval of the managers (Baddoo & Hall, 2003), (Kasse & McQuaid, 2000), commercial pressure where the product owner should be able to prioritize conflicting user requirements (Baddoo & Hall, 2003), (Kasse & McQuaid, 2000), and support tools that will enable better decision (Kasse & McQuaid, 2000).

To analyze factors models used in the assessment of ASDs adoption, we conducted a Systematic Literature Review to collect different factors from scientific data source applying a research method oriented to this objective.

This article presents factors models for the adoption of Scrum methods with a background of agile adoption models. The article is organized as follows: Section 2 presents the systematic literature review protocol; in Section 3 the identified models are introduced; and Section 4 contains the final discussion and future work.

2. Systematic Literature Review

Kitchenham defines a systematic literature review (SLR) as the process to identify, evaluate and interpret all available research relevant to a particular research question, or phenomenon of interest (Kitchenham, 2007), establishing a sequential tailored set of activities, in Figure 1 shows the phases used in this study. In addition, primary studies are contributors of secondary studies (Kitchenham, 2007). In our case, we decided to use a SLR considering that some search results would be case studies or similar and others would be secondary studies. Consequently, we did not use words related to primary or secondary studies and instead we used others related to models factors and ASD.

Some benefits of a SLR are: i) identification of the particular research questions to be investigated; ii) identification of the desired population; iii) intervention, context and outcomes and helps in summarizing the existing research evidence and others (Kitchenham, 2007).

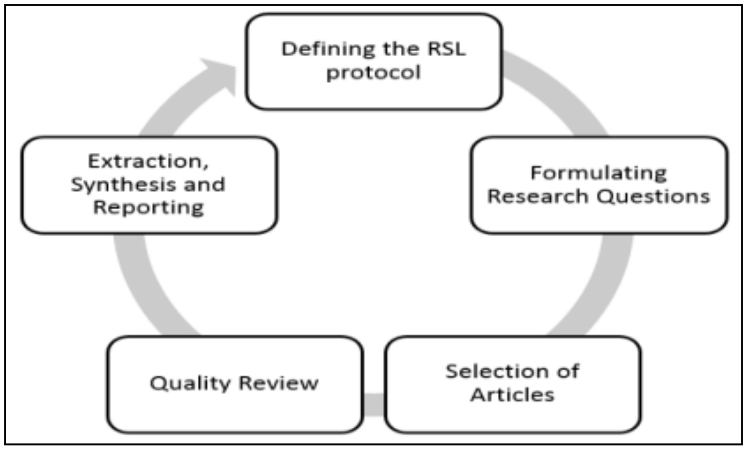

Figure 1. SLR phase, adapted from (Kitchenham, 2007)

2.1. SLR Protocol

The protocol used in this study was adapted from Kitchenham (Kitchenham, 2007) and Sulayman (Sulayman & Mendes, 2009). The phases are:

- Defining the SLR protocol: this phase is an iterative flow and covers the overall plan for the SLR process.

- Formulating research questions: in this phase the SLR research questions are identified.

- Selection of articles: in this phase inclusion and exclusion criteria are considered for selecting primary or secondary articles.

- Quality review: in this phase selected articles are submitted to a quality review, where checklists are used to validate the articles filtered in previous phase.

- Extraction, synthesis and reporting: in this phase relevant data is extracted from each article. Then, it is added, integrated and summarized in order to answer the research questions in detail.

This SLR aims to consolidate the information about factor models of agile methods adoption in the software development process. The present SRL summarizes existing models which organizes factors in order to explain the influence of these factors on the adoption of agile methods in the software development process and shows common factors shared between all identified models.

2.2. Formulating Research Question

Formulation of research question has been developed following the PICO guidelines presented by Santos (Santos, Pimenta, & Nobre, 2007): Population, Intervention, Comparison and Outcome, as described in Table 1. This research is not focused on what is the intervention compared with, therefore, it is omitted in the subsequent tables and sections. The final search string is the result of some iterations in which each string was refined according to the quantity and quality of articles.

- Population: groups of elements that are observed by the intervention. Studies presenting agile methods implementations in software processes and projects.

- Intervention: elements to be evaluated within the defined population. These are the factors models or frameworks used in the adoption of agile methods.

- Outcome: the result of the information according the investigation. Primary studies about factors models of agile methods adoption in software processes.

Keywords used in the study	
Population	(Agile OR Scrum OR XP) AND ("software process" OR Methodology OR Methods OR "software project" OR "software projects" or "software development")
Intervention	(adoption OR adopt OR implementation OR impact OR adaptation)
Outcome	(model OR framework) AND (Factor OR enable OR disable)
Research Strategy	Population AND Intervention AND Results

Table 1. *Keywords obtained from PICO strategy*

Based on the criteria defined above we established the following research question:

RQ1: Which factors models related to the adoption of ASD are used in the software industry? Models found will be used to identify relevant factors.

RQ2: What factors are common to all the identified models? Factors will be classified based on their presence in all the identified models.

The search was performed in April 2016 and includes studies before that date.

2.3. Studies Selection

The search procedure starts when the search string is built based on PICO criteria, keywords found in the studies, synonyms terms and Boolean terms AND - OR.

Once the search string was defined, this was used on the following online digital libraries to obtain primary studies (Dieste, Grimán, & Juristo, 2009), (Elberzhager, Münch, & Vi, 2012):

- ACM Digital Library (http://portal.acm.org)
- Proquest (http://www.proquest.com)
- Elsevier ScienceDirect (www.sciencedirect.com)
- Scopus (http://www.scopus.com)
- Thomson Reuters - Web of Knowledge (www.webofknowledge.com)
- IEEE (https://ieeexplore.ieee.org/)

Once the results of the search were obtained during the first iteration, titles were analyzed to determine their actual relevance. In addition, during the second iteration, abstracts and conclusions were analyzed. Both iterations were intended to identify those primary studies that answer the research questions.

Inclusion criteria refer to studies that are related to models of factors to adopt agile methods. We used: (CI1) title and abstract related to the object of study; (CI2) studies related to agile methods adoption; (CI3) studies related to Scrum adoption and (CI4) studies related to XP adoption. Exclusion criteria refer to the exclusion of studies that are not focused on factors models or agile method adoption. We used: (CE1) studies which did not relate to factors models adoption and (CE2) duplicate studies.

On the other hand, quality criteria contribute to assess the reliability of the papers. In the first iteration we applied CI1, CI2, CI3 and CI4. In the second, we applied CE1; and finally, on iteration 3, we applied CE2 (see Table 2).

Database	Results	Iterat. 1	Iterat. 2	Iterat. 3
ACM	56	15	7	2
Proquest	24	5	1	0
Science Direct	28	6	4	1
Scopus	122	41	10	6
Web of Knowledge	120	10	8	0
IEEE	3	1	0	1
Total	353	78	30	10

Table 2. *Results of studies Retrieved and Selected*

2.4. Study Quality Assessment

The checklist used to assess the quality of selected studies is listed in Table 3. These qualitative questions were obtained from Sulayma (Sulayman & Mendes, 2009). Based on the answers, each study might have the following qualifications: 1.0 if the answer is '[Y]es', 0.5 if the answer is '[P]artially' and 0.0 if the answer is '[N]o'. Thus, the study could obtain a maximum score of 9. The results of the selected studies and their partial and final scores are presented in Table 4. In our case, (Srinivasan, Dobrin, & Lundqvist, 2009) obtained 5 of 9 points (55%) and represents one of ten models identified. After an individual review and rating than 50%, we decided to include it.

Question
1. Is the methodology used suitable to address the stated research questions?
2. Does the article target the ideal population?
3. Does the article use the research methodology adequately?
4. Does the article discuss any of the previous work/literature?
5. Is the study process specified in the article repeteable?
6. Is the article oriented towards a factors framework, model or techique?
7. Do the findings address the research questions?
8. Does the article document any assumption taken?
9. Does the article document the procedure used to validate its findings?

Table 3. *Quality questions used in our study (Sulayman & Mendes, 2009).*

3. Extraction, Synthesis and Results

The models found in the articles selected as primary studies are described in this Section.

Categories are the same used by Shahane: Organization, Project, Process and People which are based on the five axis polar charts as suggested by Boehm and 3-factor comparison: People, Process, Projects (Shahane, Jamsandekar, & Shahane, 2014). Also, we consolidate in Table 5, agile methods studied in the selected articles.

Study	1	2	3	4	5	6	7	8	9	T
(Ahimbisibwe, Cavana, & Daellenbach, 2015)	Y	Y	Y	Y	Y	Y	Y	Y	P	8.5
(Shahane, Jamsandekar, & Shahane, 2014)	P	P	Y	N	P	Y	Y	P	N	5.5
(Melo, Cruzes, Kon, & Conradi, 2013)	Y	P	P	N	Y	Y	Y	P	Y	6.5
(Lee, 2012)	P	P	P	P	P	Y	Y	N	Y	5.5
(Overhage, Sebastian, Birkmeier, & Miller, 2011)	P	Y	Y	Y	P	Y	Y	P	P	7.0
(Misra, Kumar, & Kumar, 2009)	P	P	P	Y	P	Y	Y	P	Y	6.5
(Chow & Cao, 2008)	Y	Y	Y	Y	P	Y	Y	Y	P	8.0
(Chan & Thong, 2007)	P	P	P	Y	P	Y	Y	N	Y	6.0
(Srinivasan, Dobrin, & Lundqvist, 2009)	P	N	P	Y	N	Y	Y	N	Y	5.0
(Stettina & Heijstek, 2011)	Y	Y	Y	N	P	Y	Y	P	Y	7.0

Table 4. *Quality Result.*

Study	Agile	Scrum	XP	Lean	Other
(Ahimbisibwe, Cavana, & Daellenbach, 2015)	X				
(Shahane, Jamsandekar, & Shahane, 2014)	X	X	X		X
(Melo, Cruzes, Kon, & Conradi, 2013)	X	X	X	X	
(Lee, 2012)	X	X			
(Overhage, Sebastian, Birkmeier, & Miller, 2011)	X	X			
(Misra, Kumar, & Kumar, 2009)	X				
(Chow & Cao, 2008)	X	X	X		X
(Chan & Thong, 2007)	X	X	X		X
(Srinivasan, Dobrin, & Lundqvist, 2009)	X				
(Stettina & Heijstek, 2011)	X	X			
	10	7	4	1	3

Table 5. *Study by Agile Method*

3.1. Ahimbisibwe et. al. - CSF for Software Development Projects

The study of Ahimbisibwe, Cavana, Daellenbach (Ahimbisibwe, Cavana, & Daellenbach, 2015) had the purpose to identify and categorize critical success factors (CSFs) and develop a contingency model to adjust the contrasting perspectives of traditional and agile methodologies (Ahimbisibwe, Cavana, & Daellenbach, 2015).

Based on a previous systematic literature review done by the authors, there were identified 37 CSFs for software development projects within 148 articles, and categorized into three major CSFs: organizational, team and customer factors. The contingency model increases these factors by highlighting the need to match project characteristics and project management methodology to these CSFs.

3.2. Shahane et. al.- Conceptual Framework

The framework Shahane, Jamsandekar and Shahane proposed in 2014 is based on the revision of proposed factors and existing models found in a literature review. The model is oriented to the elements: organization, people, processes and projects, considered as cornerstones (Shahane, Jamsandekar, & Shahane, 2014). In addition, the model is based in one equilateral triangle (pyramid) divided in four equilateral triangle. In this model each triangle in the framework pyramid represents a set of critical factors for the success of any project (Shahane, Jamsandekar, & Shahane, 2014).

3.3. Melo et. al. - Productivity Factors Framework

The framework of Melo et al (Melo, Cruzes, Kon, & Conradi, 2013) presented in 2012 is based on the theoretical model of effectiveness Input-Process-Outcome (IPO) of Cohen and Bailey, Yeatts and Hyten. It is a multiple-case study during six months in three large Brazilian companies, which had used agile methods for

more than 2 years. The study is focused on the main productivity factors perceived by the team members through interviews, retrospectives and documentation (Melo, Cruzes, Kon, & Conradi, 2013). As a result, it was developed a conceptual framework, using thematic analysis to understand the possible mechanisms behind these productivity factors. Agile team's management proved to be the most influential factor in the achievement of agile team productivity. For intra-team level, the main productivity factors were team design (structure and allocation of work) and employee turnover (Melo, Cruzes, Kon, & Conradi, 2013). For inter-team level, the main productivity factors were the ability to coordinate effectively through appropriate interfaces, avoiding delays of the software.

3.4. Lee - Scrum Performance Dynamic

The Dynamic Performance Scrum is a framework designed by Rich Lee in 2012, based on a qualitative research, led by observation and interviews of two teams (Lee, 2012). The categories used were covered based on the review of existing literature, such as: personal attributes, characteristics of user stories, capacity of project team, team autonomy, team diversity, change response, efficiency and performance team in software development (Lee, 2012).

3.5. Overhage et. al. - Framework of Drivers and Inhibitors to Developer Acceptance

Overhage, Schlauderer, Birkmeier framework was defined in 2011 as a group of drivers and inhibitors to the developer acceptance of Scrum. This framework is based on the Extended Technology Acceptance Model (TAM), applicable to developer acceptance of this methodology (Overhage, Sebastian, Birkmeier, & Miller, 2011).

Their results were based on six qualitative interviews applied to six Scrum experienced experts of a German company. The general determinants defined in the TAM were refined with several factors that have influence in the willingness of developers to use Scrum (Overhage, Sebastian, Birkmeier, & Miller, 2011).

3.6. Misra et. al. - Success Factors Framework

This framework was developed in 2009 and it is based on a literature review. It establishes 14 factors derived from its hypothesis, and the most important factors were determined based on questionnaires, where nine factors were related to success: satisfaction customer, customer collaboration, customer commitment, decision time, corporate culture, control, characteristics of people, culture social, and learning and training (Misra, Kumar, & Kumar, 2009). Multiple regression models were used to test the relations between the success factors (Misra, Kumar, & Kumar, 2009).

3.7. Chow and Cao - Factors Model

The Chow-Cao Model was defined in 2007 (Chow & Cao, 2008). It has the following characteristics: i) it is based on a quantitative analysis; ii) it identified 12 critical factors grouped into 4 categories of project success: quality, scope, time and cost; iii) the surveys were distributed to professionals of the agile community in 109 projects in 25 countries; iv) to validate the model they used regression techniques such as complete model and optimized model and 10 out of 48 hypotheses were verified; v) only three factors should be considered critical: delivery strategy, agile software engineering techniques and team capabilities.

3.8. Chan and Thong - Conceptual Framework

The conceptual framework of Chan and Thong was developed in 2007, based on previous empirical studies about the acceptance of Systems Development Method (SDM) in the organization/individual and studies on agile methodologies, where potential factors were identified (Chan & Thong, 2007). These factors were classified as: (i) individual factors associated with software developers; (ii) organizational factors associated with the management and organization; (iii) factors associated with agile methodology; and (iv) factors associated with the relationship creator-client (Chan & Thong, 2007).

3.9. Srinivasan et. al.– Technical factors Framework

In this paper, Srinivasan, Dobrin and Lundqvist (Srinivasan, Dobrin, & Lundqvist, 2009) have found that there are technical issues (requirement management, and testing), as well as organizational issues (process tailoring, knowledge sharing and transfer, culture change and infrastructure support). They were considered as a framework for preliminary guidance.

3.10. Stettina and Heijstek. Five Agile Factors.

This study is based on the qualitative model of Moe et al., Stettina and Heijstek (Srinivasan, Dobrin, & Lundqvist, 2009), who developed a quantitative questionnaire organized among five dimensions of agile teamwork, analogous to the Five Factor Model in contemporary psychology. This survey was conducted with 79 individuals and eight international Scrum teams.

4. Final Discussion and Future Work

Some of the studies found evaluate factors related to the method itself, and others explore factors related with the organizational practice taking into consideration different factors. Consequently, there are factors, which were studied individually and others that were classified in the same category by the selected studies. Categories are the same used by Shahane: Organization, Project, Process and People.
The amount of factors per category is shown in Figure 2.

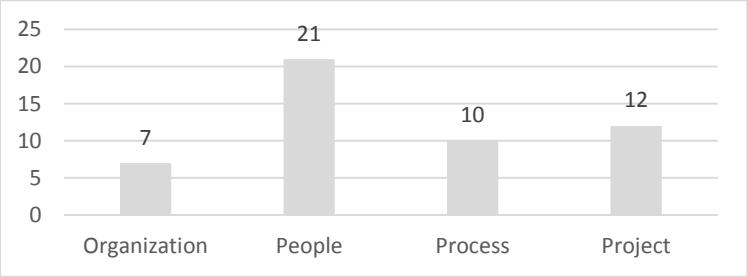

Figura 2. *Factors Quantity per Coincidence*

Then those factors are listed by article in Table 6, showing which factors have more concurrence. The higher the concurrence is better the comparison analysis between models. Only eight factors had the higher concurrence, greater than three. They are: user participation, team orientation, team qualified attributes, training and learning, teamwork, team's experience, user support, and requirements generation.

User participation and User support. Both enables software developers and customers to work towards a common objective in the most effective and agile way. Therefore, software developers are more likely to adopt agile methodology when they have a good understanding with its customers (Ahimbisibwe, Cavana, & Daellenbach, 2015), (Shahane, Jamsandekar, & Shahane, 2014), (Melo, Cruzes, Kon, & Conradi, 2013), (Misra, Kumar, & Kumar, 2009), (Srinivasan, Dobrin, & Lundqvist, 2009), (Chow & Cao, 2008), (Chan & Thong, 2007).

Team orientation. Team goals are prioritized over individual goals. This increases individual responsibility (Ahimbisibwe, Cavana, & Daellenbach, 2015), (Melo, Cruzes, Kon, & Conradi, 2013), (Lee, 2012), (Overhage, Sebastian, Birkmeier, & Miller, 2011), (Chan & Thong, 2007), (Srinivasan, Dobrin, & Lundqvist, 2009).

Team qualified attributes. The technical and business skills are the most relevant (Lee, 2012), (Misra, Kumar, & Kumar, 2009), (Chow & Cao, 2008), (Chan & Thong, 2007). According to the authors, agile method is best adopted by developers when they have received adequate training.

Training and learning. Shared knowledge that allows its maintenance (Ahimbisibwe, Cavana, & Daellenbach, 2015), this allows an organization to develop knowledge and to be better prepared to implement the methodology (Ahimbisibwe, Cavana, & Daellenbach, 2015), (Shahane, Jamsandekar, & Shahane, 2014), (Lee, 2012), (Chan & Thong, 2007)

Teamwork. A work environment based on collaboration and an accurate social pressure; these elements will allow the developer an early adoption of the agile method (Melo, Cruzes, Kon, & Conradi, 2013), (Lee, 2012), (Overhage, Sebastian, Birkmeier, & Miller, 2011), (Misra, Kumar, & Kumar, 2009). When developers work in a communicative, dynamic and progressive environment, they are prepared for success (Misra, Kumar, & Kumar, 2009).

In the people category, coincidentally, the most relevant factors are oriented to the team, team attributes, teamwork and team experience (Ahimbisibwe, Cavana, & Daellenbach, 2015), (Melo, Cruzes, Kon, & Conradi, 2013), (Stankovic, Nikolic, Djordjevic, & Cao, 2013), (Lee, 2012), (Overhage, Sebastian, Birkmeier, & Miller, 2011), (Misra, Kumar, & Kumar, 2009), (Chow & Cao, 2008), (Chan & Thong, 2007), (Srinivasan, Dobrin, & Lundqvist, 2009), which allow the team to be empowered.

Requirements generation. The functional specification of the final product is very important. It should be consolidated in an iterative way with continued participation of the user. A good definition will allow proper implementation (Shahane, Jamsandekar, & Shahane, 2014), (Lee, 2012), (Chan & Thong, 2007), (Srinivasan, Dobrin, & Lundqvist, 2009).

Category	Factor	Ahimbisibwe et. al.	Shahane et. al.	Melo et. al.	Lee	Overhage et. al.	Misra et. al.	Chow and Cao	Chan and Thong	Srinivasan et. al.	Stettina and Heijstek	Coincidence
People	User participation	X	X				X	X	X	X		6
People	Team orientation	X		X	X	X			X		X	6
People	Team qualified attributes				X		X	X	X			4
People	Training and Learning	X	X				X				X	4
People	Teamwork			X	X	X	X					4
People	Team's Experience	X	X		X				X			4
People	User support	X		X				X		X		4
Process	Requirements generation		X		X				X	X		4
People	Work Environment						X	X	X			3
People	Autonomy					X				X	X	3
Process	Simplicity		X					X	X			3
Process	Methodology		X			X		X				3
Process	Process Adaptation					X			X	X		3
Project	Complexity	X	X		X							3
Project	On-Site / Off * Shore		X		X				X			3
Project	Project Size	X	X				X					3
Project	Urgency	X	X		X							3
Organization	Management Support		X							X		2
Organization	Vision and mission	X	X									2
Organization	Culture Change		X							X		2
Organization	Shared Leadership	X									X	2
People	Team Size		X	X								2
People	Diversity team			X	X							2
People	Efficiency				X				X			2
People	Communication	X							X			2
People	Team composition	X		X								2
Process	Knowledge, sharing & transfer					X				X		2
Process	Dynamism		X					X				2
Process	Maturity					X			X			2
Project	Project planning and controlling	X				X						2
Project	Technological uncertainty	X								X		2
Organization	Security		X									1
Organization	Stability		X									1
Organization	Organizational culture	X										1
People	Learning curve				X							1
People	Career opportunities								X			1
People	Transparency					X						1
People	Staff turnover			X								1
People	Location			X								1
People	Team's expertise	X										1
People	Customer experience	X										1
Process	Perception of Use								X			1
Process	Redundancy										X	1
Process	Testing Approaches									X		1
Project	Change management	X										1
Project	Cost				X							1
Project	Client satisfaction						X					1
Project	Rules								X			1
Project	Project Specification Changes	X										1
Project	Project criticality	X										1

Table 6. *Factors per Author*

The appearance of these models came from 2007 to 2015, models Chan – Thong (Chan & Thong, 2007) and Chow – Cao (Chow & Cao, 2008) are the oldest and the most recent is Ahimbisibwe et.al.
Table 7 indicates the study technique (research) that was used for the selection of factors, which were corroborated using questionnaires, surveys or using empirical and analytical case studies.

Study	Literature review	Questionaires	Case study	Exploratory Expert Opinion
(Ahimbisibwe, Cavana, & Daellenbach, 2015)	X			
(Shahane, Jamsandekar, & Shahane, 2014)			X	
(Melo, Cruzes, Kon, & Conradi, 2013)			X	
(Lee, 2012)				X
(Overhage, Sebastian, Birkmeier, & Miller, 2011)	X	X		
(Misra, Kumar, & Kumar, 2009)	X	X		
(Chow & Cao, 2008)	X		x	
(Chan & Thong, 2007)	X			
(Srinivasan, Dobrin, & Lundqvist, 2009)	X			
(Stettina & Heijstek, 2011)		X		

Table 7. *Characteristic of Models Found*

Finally, it was found empirical work about impact factors analysis in the adoption of agile methods, such as Mann (Mann & Maurer, 2005), Imreh (Imreh & Raisinghani, 2011), Kumar (Kumar & Kumar Bhatia, 2012) Pagrut (Pagrut, 2008), Mark (Mark, 2011), Stankovic (Stankovic, Nikolic, Djordjevic, & Cao, 2013), among others. Those articles may be considered for future research related to impact factors. In addition, as a future research, we will consider a case study to evaluate those models in a local context.

Acknowledgment

This work is framed within ProCal-ProSer Contract N° 210-FINCYT-IA-2013 (Innovate Perú): “Productivity and Quality Relevance Factors in small software development or service organizations adopted ISO standards”, the Software Engineering Development and Research Group and the Department of Engineering of Pontificia Universidad Católica del Perú.

References

Abrahamsson, P., Salo, O., Ronkainen, J., & Warsta, J. (2002). Agile Software development methods. Review and analysis. Finland: VTT Publications.

Agile Alliance. (06 de 04 de 2016). What is Agile Software Development? Obtenido de https://www.agilealliance.org/agile101/what-is-agile/

Agile Manifesto. (2001). Recuperado el 8 de Aug de 2015, de http://agilemanifesto.org

Ahimbisibwe, A., Cavana, R. Y., & Daellenbach, U. (2015). A contingency fit model of critical success factors for software development projects: A comparison of agile and traditional plan-based methodologies. Journal of Enterprise Information Management, 28(1), 7 - 33.

Azevedo Santos, M. d. (2011). Agile Practices: An Assessment of Perception of Value of Professionals on the Quality Criteria in Performance of Projects. Journal of Software Engineering and Applications - Scientific Research, 4(12), 700-709.

Baddoo, N., & Hall, T. (Apr de 2003). De-motivators for Software Process Improvement: An Analysis of Practitioners' Views. The Journal of Systems and Software, 66(1), 23–33.

Bohem, B. (Jan de 2002). Get ready for agile methods, with care. Computer, 35(1), 64-69.

Cao, L., Mohan, K., Xu, P., & Ramesh, B. (August de 2009). A framework for adapting agile development methodologies. European Journal of Information Systems, 18(4), 332–343.

Chan, K., & Thong, J. (2007). An Integrated Frame-work of Individual Acceptance of Agile Methodologies. Pacific Asia Conference on Information Systems (PACIS) (pág. Paper 154). Auckland, New Zealand: AISEL.

Chow, T., & Cao, D.-B. (Jun de 2008). A survey study of critical success factors in agile software projects. Journal of Systems and Software, 81(6), 961-971.

Conboy, K., & Fitzgerald, B. (June de 2010). Method and developer characteristics for effective agile method tailoring: A study of XP expert opinion. ACM Trans. Softw. Eng. Methodol, 20(1), Paper 2, 30p.

Daghfous, A., & White, G. R. (1994). Information and innovation: a comprehensive representation. Research Policy, 23(3), 267-280.

Dieste, O., Grimán, A., & Juristo, N. (2009). Developing search strategies for detecting relevant experiments. Empirical Software Engineering, 14(5), 513-539.

Dingsøyr, T., Nerur, S., Balijepally, V., & Brede Moe, N. (Jun de 2012). A decade of agile methodolo-gies: Towards explaining agile software development. Journal of Systems and Software, Elsevier, 85(6), 1213–1221.

Dubakow, M. (10 de 2010). 10 Most Common Mistakes in Agile Adoption. Recuperado el 06 de 04 de 2016, de Targetprocess Inc: https://www.targetprocess.com/blog/2010/10/10-most-common-mistakes-in-agile-adoption-part-i/

Dyba, T. (Set-Oct de 2000). Improvisation in Small Software Organizations. IEEE Software, 17(5), 82–87.

Dyba, T. (May de 2005). An empirical investigation of the key factors for success in software process improvement. IEEE Transactions on Software Engineering, 31, 410-424.

El Emam, K., Goldenson, D., McCurley, J., & Herbsleb, J. (1998). Success or Failure? Modeling the Likelihood of Software Process Improvement. International Software Engineering Research Network, ISERN-98-15. ISERN-98.

Elberzhager, F., Münch, J., & Vi, T. (Jan de 2012). A systematic mapping study on the combination of static and dynamic quality assurance techniques. Information and Software Technology, 54(1), 1-15.

Fichman, R. G., & Kemerer, C. F. (1999). The Illusory Diffusion of Innovations: An Examination of Assimilation Gaps. Information Systems Research, 10(3), 255 - 275.

Goodman, P. (1996). The Practical Implementation of Process Improvement Initiatives. En N. Fenton, R. Whitty, & Y. Lizuka (Edits.), Software Quality Assurance and Measurement: A Worldwide Perspective (pág. 315). International Thomson Computer Press.

Hajjdiab, H., & Taleb, A. (Sep de 2011). Adopting Agile Software Development: Issues and Challenges. International Journal of Managing Value and Supply Chains (IJMVSC), 2(3).

Hass, K. B. (10 de March de 2009). The Blending of Traditional and Agile Project Management. Recuperado el 06 de September de 2015, de PM Times for Projects Managers: https://www.projecttimes.com/articles/the-blending-of-traditional-and-agile-project-management.html.

Highsmith, J., & Cockburn, A. (Sep de 2001). Agile software development: the business of innovation. Computer, 34(9), 120-127.

Imreh, R., & Raisinghani, M. S. (Oct de 2011). Impact of Agile Software Development on Quality within Information Technology Organizations. Journal of Emerging Trends in Computing and Information Sciences, 2(10).

Kanane, A. (2014). Challenges related to the adoption of Scrum. Case study of a financial IT company. UMEA University, Department of informatics. IT management master program. UMEA University.

Kasse , T., & McQuaid, P. A. (Aug de 2000). Factors Affecting Process Improvement Initiatives. Crosstalk: the Journal of De-fense Software Engineering, 13(8), 4-8.

Kitchenham, B. (2007). Guidelines for performing Systematic Literature Reviews in Software Engineering. EBSE Technical Report. EBSE-2007-01, Keele University , School of Computer Science and Mathematics.

Kumar, G., & Kumar Bhatia, P. (2012). Impact of Agile Methodology on Software Development Process. International Journal of Computer Technology and Electronics Engineering (IJCTEE), 2(4), 46-50.

Lee, R. (2012). The Success Factors of Running Scrum: a Qualitative Perspective. Journal of Software Engineering and Applications, 5(6), 367-374.

Mann, C., & Maurer, F. (2005). A Case Study on the Impact of Scrum on Overtime and Customer Satisfaction. Agile Development Conference (ADC'05), (págs. 70-79).

Mark, P. (2011). On empirical research into scrum. Institute for Software Research Carnegie Mellon University, Pittsburgh. Obtenido de http://www.cs.cmu.edu/~mcp/agile/oersa.pdf

Melo, C., Cruzes, D. S., Kon, F., & Conradi, R. (Feb de 2013). Interpretative case studies on agile team productivity and management. Information and Software Technology, 55(2), 412–427.

Mishra, D., & Mishra, A. (Jan de 2011). Complex software project development: agile methods adoption. Journal of Softwrae Maintenance and Evolution: Research and Practice, 549–564.

Misra, S. C., Kumar, V., & Kumar, U. (Nov. de 2009). Identifying some important success factors in adopting agile software development practices. Journal of Systems and Software, 82(11), 1869-1890.

Niazi, M., Wilson, D., & Zowghi, D. (2006). Critical Success Factors for Software Process Improvement Implementation: An Empirical Study. Software Process: Improvement and Practice, 11(2), 193-211.

Overhage, S., Sebastian, S., Birkmeier, D., & Miller, J. (2011). What Makes IT Personnel Adopt Scrum? A Framework of Drivers and Inhibitors to Developer Acceptance. 44th Hawaii International Conference on System Sciences (HICSS), (págs. 1-10). Kauai.

Overhage, S., Sebastian, S., Birkmeier, D., & Miller, J. (2011). What Makes IT Personnel Adopt Scrum? A Framework of Drivers and Inhibitors to Developer Acceptance. 44th Hawaii International Conference on System Sciences (HICSS), (págs. 1-10). Kauai.

Oyeyipo, E. (2011). An empirical study of requirements management in an agile-Scrum development enviroment. Thesis Master of Sciemce, Universidad San Marcos, Department of Computer Science, Texas.

Pagrut, D. (2008). The Impact of an Agile Scrum on Software Testing: A Case Study of Tech Mahindra Limited. 5th International Conference On Software Testing. STeP-IN SUMMIT.

Powell, T. C. (1995). Total Quality Management as Competitive Advantage: A Review and Empirical Study. Strategic Management Journal, 16, 15-37.

Repenning, N. P., & Sterman, J. D. (Summer de 2001). Nobody Ever Gets Credit for Fixing Problems that Never Happened: Creating and Sustaining Process Improvement. California Management Review, 43(4), 64-88.

Rogers, E. M. (2003). Diffusion of Innovations (Fifth Edition ed.). New York: The Fnree Press.

Rousseau, D. M., & McCarthy, S. (Mar de 2007). Educating Managers From an Evidence-Based Perspective. Academy of Management Learning & Education, 6(1), 84-101.

Santos, C., Pimenta, C., & Nobre, M. (2007). A estratégia PICO para a construção da pergunta de pesquisa e busca de evidências. Revista Latino-Americana de Enfermagem, 15(3), 508-511.

Shahane, D., Jamsandekar, P., & Shahane, D. (2014). Factors influencing the agile methods in practice - Literature survey & review. International Conference on Computing for Sustainable Global Development (INDIACom), (págs. 556-560). New Delhi.

Srinivasan, J., Dobrin, R., & Lundqvist, K. (2009). 'State of the Art' in Using Agile Methods for Embedded Systems Development. 33rd IEEE International Annual Conference on Computer Software and Applications, (págs. 522-527). Seattle.

Stankovic, D., Nikolic, V., Djordjevic, M., & Cao, D.-B. (June de 2013). A survey study of critical success factors in agile software projects in former Yugoslavia IT companies. Journal of Systems and Software, 86(6), 1663-1678.

Stettina, C., & Heijstek, W. (2011). Five Agile Factors: Helping Self-management to Self-reflect. 18th European Conference on Systems, Software and Service Process Improvement, EuroSPI 2011, (págs. 84-96). Roskilde: EuroSPI .

Sulayman, M., & Mendes, E. (2009). A Systematic Literature Review of Software Process Improvement in Small and Medium Web Companies. International Conference on Advanced Software Engineering and Its Applications, ASEA. 59, págs. 1-8. Jeju Island: Springer.

Zhang, X., Hu, T., Dai, H., & Li, X. (2010). Software Development Methodologies, Trends, and Implications. Southern Association for Information Systems Conference (pág. Paper 31). SAIS

Notas Biográficas:

Marilyn Sihuay es Titulada y Colegiada en Ingeniería de Sistemas, con más de 17 años de experiencia como Project Manager, Test Manager, Analista de Calidad de software y Analista de Sistemas. Con amplios conocimientos en Procesos de Implementación y Aplicación de Mejores Prácticas en SCRUM, ITIL, ISTQB, CMMI, Gestión de Proyectos PMP, ISO 9001:2000, Cloud Computing. Líder de proyectos clásicos y ágiles, de mejora continua y desarrollo de software, con amplio manejo del presupuesto, estrategia, planificación, ejecución y cierre; cumpliendo los objetivos del proyecto con éxito en tiempo y calidad, compromiso, honestidad, perseverancia y sentido de urgencia.

Abraham Dávila es investigador y profesor principal de la Pontificia Universidad Católica del Perú (PUCP) desde el 2000. Dirige y es investigador principal del proyecto ProCalProSer (2013-2016 Fase I y 2017-2018 Fase II) y miembro fundador de GIDIS-PUCP. Posee el grado de bachiller en ciencias con mención en Ingeniería Mecánica y magister en Informática por la PUCP. Miembro del grupo de trabajo de la ISO/IEC que elabora la norma ISO/IEC 29110. Sus principales áreas de interés son calidad en informática (a nivel de proceso software, productos y gestión de servicios) y educación en ingeniería de software.

Marcelo Pessoa es Ingeniero Electrónico, tiene una maestría, doctorado y libre docencia por la Universidad Politécnica de San Pablo – Brasil. Profesor del Dpto. de Ingeniería de la Producción desde 1987. Tiene experiencia e investigaciones en las áreas de sistemas de operaciones, computación, electrónica, telecomunicaciones y automatización. Miembro de la Comisión del Estudio de

Procesos del Ciclo de Vida de Software de la ABNT en el área de Ingeniería de Software para la elaboración de normas nacionales e internacionales en la ISO. Coordinador del CEGPTI Curso de Especialización en Gestión de la TI desde 2008. Coordinador del curso Análisis de Negocio basado en BABOK. Fue Director-Presidente de la Fundación Carlos Alberto Vanzolini en el periodo 2002-2005 y después miembro del Consejo Curador de las misma Fundación. Actualmente es vice-Jefe del Dpto. de Ingeniería de la Producción 2015/2017. Trabaja como investigador en los laboratorios eLabSoft donde realiza investigación sobre Fábrica de Software y Proceso Software. También es investigador de LADOS (Laboratorio de Análisis, Desarrollo y Operaciones de Sistemas donde desarrolla investigación sobre sistemas tecnológicos avanzados, combinando software y servicios tecnológicos para la generación de innovaciones, desarrollo de nuevos productos y servicios tecnológicos para la re-estructuración de los procesos productivos.