

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/320625185>

# Challenges in Migration from Waterfall to Agile Environments

Article · September 2017

DOI: 10.13189/wjcat.2017.050302

CITATIONS

3

READS

2,425

1 author:



**Fernando Almeida**

Instituto Superior Politécnico Gaya

103 PUBLICATIONS 224 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Innovation and Entrepreneurship [View project](#)



Observatory of Portuguese Academic Spin-offs [View project](#)

# Challenges in Migration from Waterfall to Agile Environments

Fernando Almeida

Faculty of Engineering, Institute of Systems and Computer Engineering Technology and Science, Oporto University, Portugal

Copyright©2017 by authors, all rights reserved. Authors agree that this article remains permanently open access under the terms of the Creative Commons Attribution License 4.0 International License

**Abstract** Software development methodologies are constantly evolving due to the appearance of new technologies and increasingly demanding user requests. Organizations seek to respond to these new challenges by changing their work practices, both at the technical and organizational levels. Agile methodologies thus emerge as a new development model characterized by greater flexibility and responsiveness to these new requirements, but also introduced new challenges. This study adopts a systematic review methodology to identify the key challenges in the migration process from traditional software development paradigms to agile methodologies. Four dimensions of challenges and a total of 49 sub-dimensions were found. Additionally, we discussed how those challenges can emerge, and what practices can be applied to properly manage and mitigate them.

**Keywords** Software Development, Agile, Waterfall, Challenges, Software Practices

## 1. Introduction

Software has become larger, more complex, and demands high quality standards, because it now occupies a prominent place within organizations, and not infrequently, is an essential part of the business. In addition to these factors, the dynamics of the business market and the constant emergence of new technologies, make their projects extremely dynamic and with short deadlines. These features are prevalent in all software development projects, and managing them is crucial to produce efficient and profitable systems. For this, according to several authors, the adopted software process is a critical element to achieving success [1-3].

In recent years, many initiatives are emerging to create a software production method that is more suited to current market needs, where the core word is "agility". Several models of so-called agile methods have been created, all focused on de-bureaucratizing the activities associated with software production. Among the most used agile

methodologies are SCRUM, XP, TDD, Agile Unified Process, among others. Each of these methods is usually a specialist in some software engineering practices, and when an organization wants to adopt the agile ecosystem, the correct approach is to implement a set of methods so that its entire development process can benefit from those practices.

Unlike conventional methodologies, inspired by the waterfall model, the new agile approaches to software development do not have the pretension to know all the requirements that must be developed beforehand, but rather to focus on small sets of functionalities that originate frequently released versions to the client. The paradigm breaks in the way the software is developed, allowing the absorption of new requirements and even possible changes in the requirements already developed.

There are numerous studies that focus on the implementation of agile methods in corporate environments with the aim of streamlining and simplifying existing processes in the organizational workplace environment. However, before deploying it, is necessary to think on some problems that will certainly be faced during this process of migration and change of the development paradigm. The current studies focus exclusively on two aspects. In the first one, there are scientific studies that adopt quantitative and qualitative methodologies to identify the challenges and difficulties experienced by companies in this migration process to agile. At another level, we find essentially opinion articles that share the experiences faced by IT companies. However, there is the absence of studies that synthesize and group these challenges for each business areas. This paper is organized as follows: we initially perform a revision of literature in the field of adoption agile software practices. After that initial phase, we present the adopted methodology and we identify the main categories of challenges. Then, we look in detail for each dimension of challenges and we discuss how those challenges can be identified, managed and mitigated. Finally, the conclusions of this work are drawn.

## 2. Literature Review

In the literature, we can already find several studies that

address the differences between the traditional development approaches (e.g., waterfall) and agile methodologies. The presentation of the basic agile principles, how the agile software development processes work, analyze the differences and similarities between waterfall and agile are common approaches [4-6]. One of the most comprehensive studies is a systematic review of empirical studies of agile. It has been possible to identify until 2008 the existence of approximately 2000 scientific studies that address the issue of Agile software development [7]. The study [8] uses a qualitative approach based on interviews of the industry expertise to emphasize the difference between those two software development approaches.

We can also find some manuscripts that look to the challenges and difficulties in the migration process for agile methodologies. The challenges are classified into four main categories [9]: (i) organization and management; (ii) people; (iii) process; and (iv) tools. These challenges may change in different organizations, but they should be properly analyzed and mitigated [10]. The study [11] identified three main groups of challenges: (i) culture; (ii) team practices; and (iii) scaling. The work [12] explores the factors limiting the implementations of agile practices in the software industry. It was identified three main factors: (i) high dependency on people/personnel; (ii) organizational dependency; and (iii) high impact on organizational structure and culture. The work [13] perform a review of requirement engineering issues and challenges in various software development methods. Within agile software requirements, the following challenges were identified: (i) inadequate identification of formal requirements; (ii) missing requirements, which are only discovered in next iteration; (iii) conflicting and ambiguous requirements; (iv) poor emphasis on non-functional requirements (NFR) modeling; and (v) inadequate requirement change management. The study [14] concludes that the main challenges when implementing an agile software development is to respond to the changing needs or requirements, thereby satisfying the customer needs rather than following a specific set of practices. User involvement is another challenge that was evidenced in Extreme Programming projects [15].

There are also studies that look to the challenges of adoption agile approaches in specific fields of activity. The adoption of agile methods has been slow in the public sector, due to the following reasons: (i) documentation, (ii) personnel education, experience and commitment; (iii) stakeholder communication and involvement; (iv) roles in an agile set-up; (v) location of agile teams; (vi) legislation; and (vii) complexity of SW architecture and system integration [16]. The banking sector is another traditional sector of activity that is slightly adopting agile. The study [17] identifies three main challenges: (i) cross-team dependencies; (ii) reporting and tracking of projects; and (iii) assure end-to-end quality.

The study of best practices in the adoption of agile

methodologies is also another research topic. Complex adaptive systems (CAS) theory can be used for better understanding how agile software development practices can be exploited using the three dimensions (product, process and people) of CAS [18]. The study [19] looks into the factors leading to a successful and failure adoption of agile projects implemented in traditionally waterfall environments. Six constructs factors were recognized: (i) culture; (ii) customer involvement and mandate; (iii) stakeholder involvement and buy-in; (iv) team structure and team logistics; (v) project type and project planning; and (vi) skill level and attitude of the team members. The practical implementation of agile has diverged in some situations from the original idea described in the Agile Manifesto, which may limit the realization of agile's benefits [20].

There are also studies that compare several case studies in the area of migration to agile environments. The effectiveness of agile methodologies can be better in certain environments and for specified projects [21]. The study [22] analyses the successful transition to agile developments faced by Primavera Systems. The study [23] presents a case study for agile adoption in a government entity in the UAE. These results are compared and analyzed with the outcomes obtained from other published case studies in this domain. The study [24] used a survey to identify the main issues in implementing SCRUM using two companies from Pakistan as case studies. The findings identify several issues, such as: quality items pileup, module integration issues, code quality, disruption in team work, mature vs. immature scrum, sprint duration, lack of Scrum training, release process, backlog management, no technical practices, multiple teams, metrics, risk management, documentation and over idealistic.

The process of development using agile methodologies is also composed by several challenges. The study [25] points out the issues and challenges discovered through an in-depth case study in a company which has employed Scrum for many projects. The dimension of people in organizations is considered a key challenge in agile development [26]. Additionally, adoption of agile methods requires a dramatic change in the organization. Team managers in agile environments play different roles and have different responsibilities, which can affect the whole software development process [27].

There are other studies analyze the impact of adopting agile methodologies in a business context. The study [28] explores the positive and negative changes that have occurred in a Silicon Valley leader company in mobile data synchronization and management software systems. The work [29] uses a case study at Portbase to identify the main challenges in the transition from waterfall to SCRUM. The study [30] looks into the benefits of adopting agile methodologies on large and distributed projects. Evidences extracted from a case study confirmed that agile software development methodologies perform also better than traditional approaches in those environments.

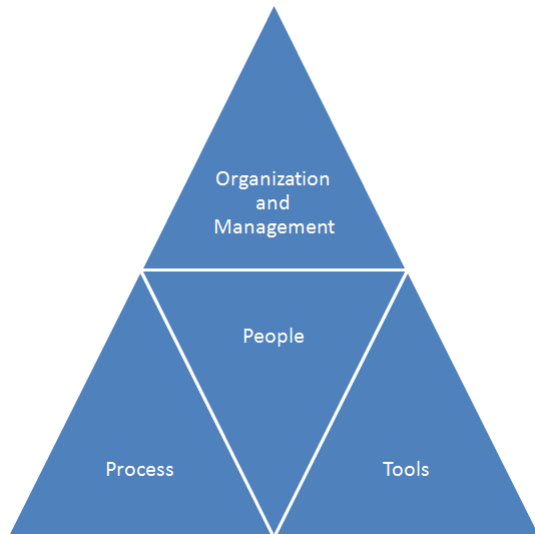
It is possible to conclude that the existing studies in the

field explorer in an asymmetric way the existing challenges in the migration process from waterfall to agile model. Some studies explore in greater depth the dimension of business organization and management practices. Others analyze in more detail the difficulties experienced by the various employees in incorporating this change in their work routines. Still other studies look to a more technical dimension, namely at the level of the processes that need reformulation and the tools that can help in this migration process. Therefore, in relation to our study, we intend to approach transversely and concisely the various challenges that exist in all of those areas.

### 3. Methodology

We adopted a systematic literature review in order to characterize and synthesize the main dimension of challenges in migration from waterfall to agile environments. This approach is suitable to identify, critically evaluate and integrate the findings of all relevant and high-quality individual studies addressing one or more research questions [31].

The first step of the process includes the creation of a conceptual diagram, where we identify the main dimension challenges of migrating to agile. The results of this step can be seen in Figure 1. A central challenge in the migration process is the people. At the top of the pyramid, we may find the organization and management, and processes and tools are two support challenges that appear at the bottom of the pyramid.



**Figure 1.** Dimension of Challenges

Then, we determine the most relevant studies published until the end of June 2017, by looking for the number of citations in Google Scholar. Table 1 shows the number total

of citations of each study. In Table 1, we only include studies that present concretely challenges and issues in moving from traditional software development models to agile approaches. The studies are ordered by the number of citations.

**Table 1.** Number of citations for each study migration from waterfall to agile environments

Study	Number of citations
Schatz & Abdelshafi, 2005 [22]	141
Serrador & Pinto, 2015 [21]	82
Hajjdiab & Taleb, 2011 [23]	24
Akif & Majeed, 2012 [24]	24
Ganesh & Thangasamy, 2012 [14]	22
Gregory et al., 2015 [11]	12
Mohammadi et al., 2009 [15]	10
Shah & Patel, 2014 [13]	6
Hamid et al., 2015 [12]	2
Gandomani et al., 2013 [9]	1
Gandomani et al., 2013 [10]	1
Nuottila et al., 2016 [16]	0
Cody et al. [17]	0

In order to focus our analysis on the most predominant studies, we needed to reduce the number of previously identified studies to allow an in-depth and comparative analysis of each of them. Therefore, a minimum cut line of six studies was considered. This number allows us to guarantee that the main challenges were found, since these studies have deserved a greater number of citations by the scientific community. In addition, we ensure the existence of at least two studies addressing each of the four dimensions of challenges.

Finally, these six studies were analyzed in detail and the identified sub-dimensions were aggregated and recorded. Next, and for each sub-dimension, the challenges identified in each were explored. Finally, a comparative table of the identified challenges is presented, in order to discover which are the most pertinent and shared by several authors.

## 4. Findings

### 4.1. Organization and Management

Firstly, we present the mind map (figure 2) of the challenges found in the "Organization and Management" dimension. Four sub-dimensions are identified: (i) legislation; (ii) culture; (iii) internal silos; and (iv) change resistance.



**Figure 2.** Mind map of "Organization and Management" dimension

**Table 2.** Classification and comparative analysis of "Organization and Management" dimension

Sub-dimension	Ref [22]	Ref [21]	Ref [23]	Ref [24]	Ref [14]	Ref [11]
Legislation						
Rules and procedures	N	N	Y	N	N	N
External audits	N	N	N	N	N	N
Culture						
Organization behavior	P	N	P	N	Y	Y
Knowledge management	N	N	N	N	P	P
Internal silos						
Departmental	N	N	P	P	N	N
Ad-hoc relationships	N	N	N	P	N	N
Change resistance						
Individual resistance	N	N	Y	N	N	N
Group resistance	N	N	Y	N	N	N
Organizational resistance	N	N	Y	N	N	N

The transition to an agile process affects all the structure of the company (e.g., development team, departments and management). Resistance can be found in all these elements, because the structure is different and people are typically used to spend a lot of time creating non-code artifacts, attending meetings, and adopting strict work processes. Therefore, they are typically oriented to plan-driven processes. Additionally, most developers in a waterfall environment present a lack of business knowledge [23].

Agile processes encourage greater participation of all employees in the organization. Knowledge sharing is one of the key elements for the success of this model. However, frequently, resistance is often encountered due to the lack of communication and ability to work in teams. Sometimes departmental silos or even ad-hoc relationships arise. Agile methodologies must be able to break these barriers to success.

Culture is also pointed out as a challenge in the process of adopting agile. Cultural barriers can appear, due to long periods of working with traditional development. However, culture is not only a barrier, but also a benefit. Software teams can enjoy major difference in their work culture resulting in collective ownership [14].

Legislation can also emerge as an issue particularly in big

projects that receive contributions from several companies and also when working with governmental organizations [16]. The rules and procedures that these organizations commonly use may be quite different. Often, it is necessary to define a rigid schedule of activities in the kickoff meeting of a project, which goes against the working principles of agile methodologies. Lastly, audit processes are essentially built to work within the context of traditional development methodologies.

Finally, table 2 identifies and summarizes the categories of challenges addressed in the most cited studies. The following nomenclature was used: "Y=yes"; "N=no"; and "P=partial". The same taxonomy was adopted for the next tables. It is possible to realize that the majority of the studies don't look in detail for the challenges related to "organization and management" dimensions.

## 4.2. People

Firstly, we present the mind map (figure 3) of the challenges found in the "People" dimension. Six sub-dimensions are identified: (i) personnel education; (ii) experience and commitment; (iii) stakeholder involvement; (iv) location; (v) training; and (vi) customer needs.



Figure 3. Mind map of "People" dimension

Lack of knowledge in agile methodologies is referred in the literature as a challenge to overcome. At this level, university institutions still choose to have curricular units too focused on traditional development paradigms. In this sense, when newcomer software engineers arrive at companies, they feel some initial difficulties in perceiving and using agile practices. This situation is mitigated by training programs, but also there are some issues, due to the lack of available courses and time consumption [23-24].

Experience and commitment are two different concepts. On the one hand, and due to the fact that agile methodologies are a novelty, it is difficult for companies to adopt a set of practices based on the experience of previous projects. On the other hand, in the agile methodologies the employees are asked to be more adaptable in order to play several roles. One of the powerful advantages of agile practices is to offer two types of responsibilities: the shared responsibility for the final product, and the individual responsibility based on each member's role [32]. A recent approach that seeks to get the best from each employee is the adoption of self-organizing teams [33]. In this approach team choose internally how to best accomplish their work, rather than being directed by others outside the team.

The difficulties of working in large and geographically distributed teams are also reported as a challenge [11]. In this dimension, it is stated that the remote location of team members and stakeholders causes agile practices to be

adjusted. Some difficulties are reported in terms of communication, different working hours, knowledge transfer, cultural differences and lack of team cohesion [34]. Solutions in terms of practices (e.g., retrospectives meetings, regular visits to co-located teams, developer/tester interactions, etc.) and adoption of software tools (e.g., Jira, Hatjitsu, Bitbucket, etc.) are suggested [35].

Finally, the involvement of stakeholders throughout the process and the correct identification of user needs are a permanent challenge for the success of agile methodologies. Priorities of user stories implementation are setup for all iterations, which offer opportunities for getting desirable results and customer satisfaction [36].

Finally, table 3 identifies and summarizes the categories of challenges addressed in the most cited studies. It is possible to realize that the "People" dimension has been the subject of considerable prominence. Only the "location" sub-dimension was mentioned by just one of the most cited studies.

#### 4.3. Process

Firstly, we present the mind map (figure 4) of the challenges found in the "Process" dimension. Nine sub-dimensions are identified: (i) team practices and roles; (ii) requirements; (iii) non-development functions; (iv) documentation; (v) cross-team dependencies; (vi) reporting and tracking; (vii) end-to-end quality; (viii) risk management; and (ix) scaling.

**Table 3.** Classification and comparative analysis of "People" dimension

Sub-dimension	Ref [22]	Ref [21]	Ref [23]	Ref [24]	Ref [14]	Ref [11]
Personnel education						
Knowledge of agile	N	N	Y	N	N	Y
Experience and commitment						
Previous projects	P	Y	Y	Y	N	Y
Adaptability	P	P	N	P	N	P
Stakeholder involvement						
Collaborative work	Y	Y	N	Y	Y	Y
Customer follow-up	Y	Y	N	N	Y	Y
Location						
Team members	N	N	N	N	N	Y
Stakeholders	N	N	N	N	N	Y
Training						
Availability of courses	N	N	Y	Y	N	N
Time consumption	N	N	P	P	N	N
Customer needs						
Identification	Y	Y	N	N	P	N
Changes	P	Y	N	N	P	N

**Figure 4.** Mind map of "Process" dimension

Team practices based on the share of the work are an intrinsic characteristic of agile methodologies. The main challenge is how making it work considering the different personalities and knowledge of agile teams. Leadership is also pointed out as a challenge, particularly in small teams, where can be difficult to find team members that have a natural affiliation to the balanced leadership model [37].

In agile paradigm, the identification of requirements is a discovering process along the way to ensure that the final requirements meet the customer's needs. Typically, requirements missing can only be an issue in a scenario where stakeholder involvement is low. On the other side, conflict requirements are a common challenge and somehow desirable in order to fulfill the customer's expectations. Besides that, not only functional requirements should be identified through the user interaction, but also non-functional functions of a project, such as portability, expansibility, security, usability, among others.

Functional working software over documentation is a principle of agile approaches. However, many times this principle is not completely understood. Documentation in agile environments is also important and the appearance of collaborative writing tools can help agile teams creating and maintaining good documentation. There are also authors that suggest the introduction of new roles (e.g., technical writer) in agile models that could be responsible to create and maintain a document management system [38].

Cross-team dependencies emerge in complex projects when many teams are working together on the same release. It can appear issues related to shared features and code, particularly when heterogeneous teams are involved. A good approach is to consider "full-stack feature teams", where teams are oriented to a product feature, originating end-to-end reduced dependencies on external teams and individuals [39].

Estimating the effort needed to deliver a given feature is one of the issues more debated in the literature. There are several approaches, but a common approach is the involvement of team commitment with the value found. Metrics are another point of debate. Velocity, defects per iteration and total project duration are commonly the most adopted metrics [22-23]. However, new metrics have been

appearing, such as business value delivered, time to market, and return on investment, among others [40].

End-to-end quality is an essential attribute of a good software development methodology. Agile development promises to provide quality assurance and it is popular for quality delivery of the project in small deliverables [41]. Several agile quality strategies can be applied, namely the use of refactoring, reviews and inspections, and the adoption of standards and guidelines [42].

Risk management is not a widely adopted practice in agile development. Consequently, there is still a wide margin of development and research. It is important to ensure their identification, prioritization, and management/control.

Finally, scaling is also a topic that has been written about and discussed by practitioners. Some issues are still present, namely: how do we guarantee the same velocity like in small projects?; how do we coordinate the dependencies in big and complex projects?; and how guarantee the same quality in big initiatives? [11]. Researchers also suggested a hybrid development method for managing large scale projects that are composed by many releases. The hybrid techniques employ agile for all new development work and waterfall to control the several releases over the time [43]. In this context, the adoption of risk management techniques becomes fundamental.

Finally, table 4 identifies and summarizes the categories of challenges addressed in the most cited studies. It is possible to identify significant differences between the importance given to each sub-dimension. The dimensions that deserve greater distinction are: requirements, cross-team dependencies, reporting and tracking, and end-to-end quality. On the other side, the least studied dimensions are: non-development functions, risk management, and scaling.

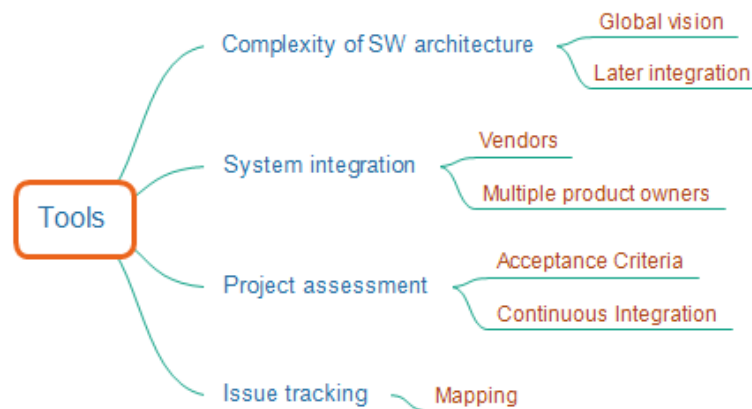
#### 4.4. Tools

Firstly, we present the mind map (figure 5) of the challenges found in the "Tools" dimension. Four sub-dimensions are identified: (i) complexity of SW architecture; (ii) system integration; (iii) project assessment; and (iv) issue tracking.



**Table 4.** Classification and comparative analysis of "Process" dimension

Sub-dimension	Ref [22]	Ref [21]	Ref [23]	Ref [24]	Ref [14]	Ref [11]
Team practices and roles						
Leadership	N	N	Y	Y	N	N
Share of work	N	N	Y	Y	N	N
Requirements						
Identification	N	Y	Y	P	Y	N
Missing	N	Y	Y	P	Y	N
Conflicting	N	Y	Y	P	Y	N
Non-development functions						
Identification	N	N	N	N	N	N
Emphasis	N	N	N	N	N	N
Documentation						
Collaborative approach	N	N	Y	Y	N	N
Cross-team dependencies						
Heterogeneity of teams	N	N	P	Y	Y	N
Full-stack feature teams	N	N	P	Y	Y	N
Reporting and tracking						
Metrics	Y	N	Y	P	N	Y
Estimation	Y	N	Y	Y	N	P
End-to-end quality						
Tests management	Y	P	N	Y	N	Y
Refactoring	N	P	N	P	N	P
Reviews and inspections	N	P	N	P	N	P
Standards and guidelines	N	P	N	P	N	P
Risk management						
Identification	N	N	N	Y	N	N
Prioritization	N	N	N	P	N	N
Management and control	N	N	N	Y	N	N
Scaling						
Velocity	N	N	N	N	N	Y
Coordination	N	N	N	N	N	Y
Quality	N	N	N	N	N	Y

**Figure 5.** Mind map of "Tools" dimension

**Table 5.** Classification and comparative analysis of "Tools" dimension

Sub-dimension	Ref [22]	Ref [21]	Ref [23]	Ref [24]	Ref [14]	Ref [11]
Complexity of SW architecture						
Global vision	P	Y	N	N	N	N
Later integration	N	Y	N	N	N	N
System integration						
Vendors	N	N	N	P	N	N
Multiple product owners	N	N	N	P	N	N
Project assessment						
Acceptance criteria	Y	N	N	N	P	N
Continuous integration	N	N	N	N	Y	N
Issue tracking						
Mapping	N	N	N	N	N	N

Technical project complexity is considered a significant issue with impact on agile project success [21]. The development of large-scale applications is particularly challenging because of their complexity and high degree of interdependency among work streams. Agile methodologies and management approaches need to overcome the risks and barriers in each development phase of a complex technical project.

The integration of systems, particularly in projects of greater complexity, also brings a set of challenges. Many times in these projects we find the need to integrate modules from several vendors. The integration of modules with different technologies is an added challenge. Furthermore, we can find multiple product owners in projects of greater complexity, which demands a clear articulation between them.

A relevant issue in an agile environment, in which the identification of requirements is a constant process along the development of the project, is the establishment of some barriers and limits to achieve the desired results. The establishment of an acceptance criterion after several iterations is one of the best practices that should be applied. Additionally, several models of project assessment have emerged in the industry. One of the models that has attracted more interest in the continuous integration, which enables organizations to frequently and reliably release new features and products. Some advantages of this practice include the reduction and increase visibility and awareness of build and test time, the detection of faults, the address of security and scalability issues in deployment pipeline, and the improvement the dependability and reliability of deployment process [44].

Finally, issues tracking can become a difficult task when they are only managed locally. It is sometimes difficult, for example, using the Scrum practice, to map an incident management process, since a detected issue in a given sprint may be associated with another sprint in a new development cycle. Integrated management of issues is a challenge that several technological tools, such as JIRA, seek to solve.

Finally, table 5 identifies and summarizes the categories of challenges addressed in the most cited studies. It is possible to conclude that "Tools" dimension is not one of the most explored areas in the literature. The main scientific studies do not address this theme, and the main source of bibliographical references in this area is the technical reports, tutorials and opinion articles written by professionals in the field.

## 5. Conclusions

Change is a complex process that involves people, organizations and processes. The introduction of agile methodologies in an organization brings with it numerous benefits deriving from the characteristics of these practices, but there is a resistance to its adoption on a large scale by the engineers and managers.

Four dimensions of challenges were identified within the context of this work. The first dimension is the people, in which we can find challenges in terms of personal education, experience and commitment, stakeholder involvement, location of teams and stakeholders, available training course and identification of customer needs. A second dimension is the organization and management, in which legal and cultural challenges arise. A third dimension is the process, in which we can find a long list of challenges, particularly the changes in terms of team practices, identification of functional and non-functional requirements, cross-team dependencies, reporting and tracking of projects, quality management, risk management, and scaling. Finally, the last dimension is related to tools, where emerge challenges associated with the technical complexity of projects, integration issues, project assessment and issues tracking.

## REFERENCES

- [1] N. Munassar, A. Govardhan. A Comparison Between Five

- Models of Software Engineering, *International Journal of Computer Science Issues (IJCSI)*, Vol. 7, No. 5, 94-101, 2010.
- [2] T. Bhuvaneswari, S. Prabakaran. A Survey on Software Development Life Cycle Models, *International Journal of Computer Science and Mobile Computing*, Vol. 2, No. 5, 262-267, 2013.
  - [3] I. Sarker, F. Faruque, U. Hossen, A. Rahman. A Survey of Software Development Process Models in Software Engineering, *International Journal of Software Engineering and its Applications*, Vol. 9, No. 11, 55-70, 2015.
  - [4] J. Juricek. Agile Project Management Principles, *Lecture Notes on Software Engineering*, Vol. 2, No. 2, 172-175, 2014.
  - [5] M. Palmqvist, M. Lapham, S. Miller, T. Chick, I. Ozkaya. Parallel Worlds: Agile and Waterfall Difference and Similarities, *Software Engineering Institute, Carnegie Mellon University, Pittsburgh*, 2013.
  - [6] A. Moniruzzaman, S. Hossain. Comparative Study in Agile Software Development Methodologies, *Global Journal of Computer Science and Technology*, Vol. 13, No. 7, 1-25, 2013.
  - [7] T. Dybä, T. Dingsoyr. Empirical studies of agile software development: a systematic review, *Information and Software Technology*, Vol. 50, 833-859, 2008.
  - [8] R. Pawar. A comparative study of agile software development methodology and traditional waterfall model, *IOSR Journal of Computer Engineering (IOSR-JCE)*, Vol. 2, No. 2, 1-8, 2015.
  - [9] T. Gandomani, H. Zulzalil, A. Ghani, A. Sultan, M. Nafchi. Obstacles in moving to agile software development methods: at a glance, *Journal of Computer Science*, Vol. 9, No. 5, 620-625, 2013.
  - [10] T. Gandomani, H. Zulzalil, A. Ghani, A. Sultan. Effective factors in agile transformation process from change management perspective, In *Proceedings of the International Conference on Advance Information Systems, E-Education & Development (CAISED 2013)*, Kuala Lumpur, Malaysia, 1-6, 2013.
  - [11] P. Gregory, L. Barroca, K. Taylor, Salah, D., Sharp, H. Agile challenges in practice: a thematic analysis, In *Proceedings of the 16th International Conference on Agile Software Development*, Helsinki, Finland, 1-13, 2015.
  - [12] S. Hamid, M. Nasir, M. Othman, R. Ahmadi. Factors limiting the implementations of agile practices in the software industry: a pilot systematic review, *Indian Journal of Science and Technology*, Vol. 8, No. 30, 1-11, 2015.
  - [13] T. Shah, S. Patel. A review of requirement engineering issues and challenges in various software development methods, *International Journal of Computer Applications*, Vol. 99, No. 15, 36-45, 2014.
  - [14] N. Ganesh, S. Thangasamy. Lessons Learned in Transforming from Traditional to Agile Development, *Journal of Computer Science*, Vol. 8, No. 3, 389-392, 2012.
  - [15] S. Mohammadi, B. Nikkhahan, S. Sohrabi. Challenges of user involvement in extreme Programming projects, *International Journal of Software Engineering and its Applications*, Vol. 3, No. 1, 19-32, 2009.
  - [16] J. Nuottila, K. Aaltonen, J. Kujala. Challenges of adopting agile methods in a public organization, *International Journal of Information Systems and Project Management*, Vol. 4, No. 3, 65-85, 2016.
  - [17] J. Cody, D. Goswami, R. Ahuja. Scaling agile at financial institutions: lessons from the trenches. Deloitte, Online available from <https://www2.deloitte.com/content/dam/Deloitte/us/Documents/financial-services/us-fsi-agile-at-scale-brochure.pdf>
  - [18] P. Meso, R. Jain. Agile Software Development: Adaptive Systems Principles and Best Practices, *Information Systems Management*, Vol. 23, No. 3, 19-30, 2006.
  - [19] M. Tanner, U. Willingh. Factors leading to the success and failure of agile projects implemented in traditionally waterfall environments, In *International Conference on Human Capital without Borders: Knowledge and Learning for Quality of Life*, Portoroz, Slovenia, 693-701, 2014.
  - [20] D. West. Water-Scrum-Fall is the reality of agile for most organizations today, *Forrester Research*, Cambridge, USA, 1-17, 2011.
  - [21] P. Serrador, J. Pinto. Does Agile work? A quantitative analysis of agile project success, *International Journal of Project Management*, Vol. 33, 1040-1051, 2015.
  - [22] B. Schatz, I. Abdelshafi. Primavera gets Agile: A Successful Transition to Agile Development, *IEEE Software*, Vol. May/June, 36-41, 2005.
  - [23] H. Hajjdiab, A. Taleb. Adopting Agile Software Development: Issues and Challenges, *International Journal of Managing Value and Supply Chains (IJMVSC)*, Vol. 2, No. 3, 1-10, 2011.
  - [24] R. Akif, A. Majeed. Issues and Challenges in SCRUM Implementation, *International Journal of Scientific & Engineering Research*, Vol. 13, No. 8, 1-4, 2012.
  - [25] J. Cho. Issues and Challenges of Agile Software Development with SCRUM, *Issues in Information Systems*, Vol. 4, No. 2, 188-195, 2008.
  - [26] K. Conboy, S. Coyle, X. Wang, M. Pikkarainen. People over Process: Key People Challenges in Agile Development Software, *IEEE Software*, Vol. 28, No. 4, 48-57, 2011.
  - [27] M. Sarpiri, T. Gandomani. How Agile managers affect the process of software development?, *IJCSNS International Journal of Computer Science and Network Security*, Vol.17 No.5, 283-286, 2017.
  - [28] J. Scarpino, R. Chicone. The quality of agile - transforming a software development company's process: a follow-up case study, *Issues in Information Systems*, Vol. 15, No. 2, 431-440, 2014.
  - [29] S. Bannink. Challenges in the transition from Waterfall to SCRUM: a case study at Portbase, In *Proceedings of the 20th Student Conference on IT*, 1-10, 2014.
  - [30] G. Papadopoulos. Moving from Traditional to Agile Software Development Methodologies also on Large, Distributed Projects, *Procedia - Social and Behavioral Sciences*, Vol. 175, 455-463, 2015.
  - [31] A. Siddaway. What is a systematic literature review and how do I do one?, Online available from <https://www.stir.ac.uk/media/schools/management/document>

- s/centregadresearch/How%20to%20do%20a%20systematic%20literature%20review%20and%20meta-analysis.pdf
- [32] I. Kotb. Mixing Roles in a Scrum Team, Online available from <https://www.frontrowagile.com/blog/posts/92-mixing-roles-in-a-scrum-team>
- [33] A. Kakar. Assessing Self-Organization in Agile Software Development Teams. *Journal of Computer Information Systems*, Vol. 57, No. 3, 208-217, 2017.
- [34] H. Kaur, H. Haddad, J. Selena. Distributed Agile Development: A Survey of Challenges and Solutions, In *Proceedings of the International Conference on Software Engineering Research and Practice*, Las Vegas, USA, 42-48, 2015.
- [35] H. Sharp, L. Barroca, A. Deshpande, P. Gregory, K. Taylor. Remote working in an Agile team, Online available from <http://agileresearchnetwork.org/wp-content/uploads/2016/09/Remoteworkingwhitepaper.pdf>
- [36] J. Abbas. Quintessence of Traditional and Agile Requirement Engineering, *Journal of Software Engineering and Applications*, Vol. 9, 63-70, 2016.
- [37] D. Fleming. Leadership balance in agile organisations, Online available from <http://www.hcamag.com/features/leadership-balance-in-agile-organisations-237747.aspx>
- [38] N. Uikey, U. Suman, A. Ramani. A Documented Approach in Agile Software Development, *International Journal of Software Engineering*, Vol. 2, No. 2, 13-22, 2011.
- [39] M. Kunal. Why You Have Dependencies, and How to Resolve Them, Online available from <https://www.thoughtworks.com/mingle/scaled-agile/2015/11/20/resolving-cross-team-dependencies.html>
- [40] P. Ifra, J. Bajwa. Metrics of Scrum Methodology, *International Journal of Computer Applications*, Vol. 149, No. 2, 24-27, 2016.
- [41] M. Sirshar, F. Arif. Evaluation of Quality Assurance Factors in Agile Methodologies, *International Journal of Advanced Computer Science*, Vol. 2, No. 2, 73-78, 2012.
- [42] S. Ambler. Agile Testing and Quality Strategies: Discipline Over Rhetoric, Online available from <http://www.ambysoft.com/essays/agileTesting.html#EndOfLifecycleTesting>
- [43] H. Saeedal, F. Arif, N. Mehmood, M. Humayun. Agile Scalability for Large Scale Projects: Lessons Learned, *Journal of Software*, Vol. 10, No. 7, 893-903, 2015.
- [44] M. Shahin, M. Babra, L. Zhu. Continuous integration, delivery and deployment: a systematic review on approaches, tools, challenges and practices, *IEEE Access*, Vol. 3, 1-12, 2017.