

An Industrial Case Study for Scrum Adoption

Hassan Hajjdiab and Al Shaima Taleb and Jauhar Ali
 College of Engineering and Computer Science
 Abu Dhabi University
 Abu Dhabi, UAE, P.O.Box 59911
 hassan.hajjdiab@adu.ac.ae

Abstract—Agile software development methods and techniques have been demonstrated to improve employee morale, enhance end-product quality, and achieve customer satisfaction. Agile adoption always comes with special challenges and thus fundamental organizational changes are necessary for successful outcome. In this paper we present an in-depth case study for adopting Scrum in a multi-team and multi-project situation to a government entity in the United Arab Emirates (U.A.E). While there are many successful stories for adopting Scrum in large and small organizations, there is little known about adoption stories in a multi-project/multi-team environments. In this paper we present a case study for an unsuccessful adoption of Scrum in a government entity in the U.A.E. The government entity under study have used the traditional waterfall approach for many years and decided to move to Agile software development method. This study highlights the challenges faced during the adoption of Scrum method with recommendations to address them. Each organization has its unique properties and one case study is not enough to draw conclusions for the U.A.E environment. However, the results obtained from this paper help software engineers to better understand the development environment and context factors for software projects in the U.A.E and many lessons may be learned for future Agile adoption.

Index Terms—Agile software development, Scrum, Software Engineering, case study;

I. INTRODUCTION

More and more organizations are moving toward adopting Agile software development [1], [2], [3], [4]. This is driven by the constant need of producing better, faster and cost-effective software solutions and at the same time maintain a high rate of employee job satisfaction [5]. An adoption attempt of Agile methods was carried out by a government entity in the UAE. This entity, referred to as *S*, consists of an IT and Telecommunication Department that includes around 200 employees. Entity *S* consists of a System Development Division that has over 50 employees, the organization structure of *S* is shown in Figure 1. The employee s roles include head of division, branch managers, project managers, system analysts, developers, technical writers and tester.

Most of entity *S* information systems and programs are developed in-house by the System Development Division. The main technologies using in developing the software system is Oracle and .NET technologies. Some Commercial Off-the-Shelf software products (COTS) are also utilized by the division. The development method that is used in *S* is a customized waterfall method. The phases of this method are:

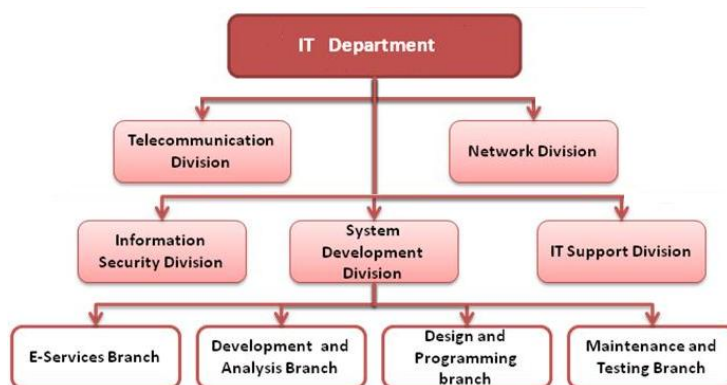


Fig. 1. Entity *S* organizational structure

- Requirement gathering
- Analysis
- Design
- Programming
- Testing
- Deployment
- Maintenance

All the units under the *Systems Development Division* comply with the waterfall model approach. The software development teams under this division experience many issues and challenges during the software development cycle, the main issues are:

- Failure of some projects
- Failure to meet deadlines
- Variations in information systems and programs quality
- Lack of knowledge transfer between team members

In 2009 a task force was formed to discuss issues and concerns faced during the software development process. The task force included representatives from every unit under the division. The main objectives is to target the above issues and come up with recommendations and remedy actions. The task force investigated alternative approach for software development process and recommended a workshop to investigate the Agile approach. A consultant, who is an expert in Agile software development, was hired. The consultant performed a thorough analysis of the currently used waterfall model, he also conducted a series of meetings with the management

and with the task force. The recommendation was to adopt Scrum approach. A training course *f* Introduction to Agile Development *f* on Scrum was offered to 30 team members from different level and roles. The course was offered under the supervision of the consultant. A survey were conducted immediately after *f*nishing the course, there was a consensus among the team members that Scrum would be better option than the current waterfall method. At the end of the training course all members of the team members were enthusiastic and optimistic and were looking forward to lead the change in entity *S*. In 2010 and based on the feedback from the task force, entity *S* management decided to adopt Scrum.

In this paper we present an industrial case study for adoption of Agile software development technologies for a government entity in the U.A.E. Case studies exhibits factors that may not be predictable for formal experiments [6], [7]. In software engineering case studies are essential to build evidence and draw conclusions on software engineering tools and methods [8]. However each corporate has unique setup, team background, cultural and project properties. This makes generalization of a case study a difficult question [9]. To evaluate the credibility of the theory behind a software engineering method, a number of case studies in different conditions must be conducted [8].

The main contribution of this paper is that, to the best of our knowledge, it is the *f*irst study on adopting an agile software engineering method in the U.A.E in a multi-team/multi-project environment. Second this paper exhibits challenges for agile adoption that are unique to the U.A.E environment and not reported in any of the studies in this domain.

The rest of the paper is organized as follows: section II presents an overview of the development and context factors of the project, section III discusses the challenges faced during the adoption process, section IV shows the proposed solutions and *f*inally section V is the conclusion.

II. DEVELOPMENT AND CONTEXT FACTORS

The result of a case study is correlated with environment in which the study was carried out [10]. Thus one cannot generalize a case study, such generalization is subject to the question of external validity [9]. In our paper we record the context factors of *S* to fully understand the generality and the utility of the *f*indings. It is also helpful to understand the differences between the challenges faced in the environment of *S* and the challenges faced in other working environments. In this study we present the development factors and the context factors of the project in *S*. The development factors studied in this paper are as described in [11] are: Personnel, Dynamism, Culture, Size and Criticality. The context factors presented are as introduced in [12] to be: Sociological, Project-specific, Technological, Ergonomics and Geographic factors. In this study six sources of data and information are used as shown in Table I

A. Developmental factors

The developmental factors of the development environment of entity *S* is shown in Figure 2. The polar graph indicates

Source	Key
Development leader	DL
Observation	OB
Project tracking	PT
Source code	SC
Employee s record	ER
Interviews	IN

TABLE I
DATA SOURCE KEY

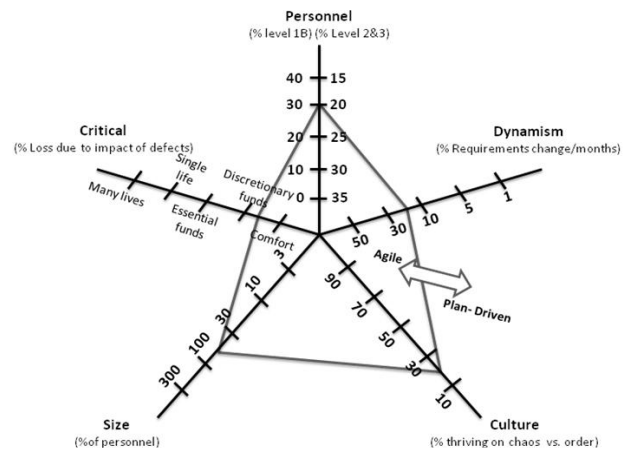


Fig. 2. Entity S developmental factors

that *S* is more traditional and plan-driven than agile. It can be clearly noticed from the graph that the percentage of loss due to impact of defects is very low due to the fact that all their projects are funded by the government entity. Also, the percentage of requirements change per month is very high due to that fact that internal customers, departments on the government entity, do not have to pay the cost of requirements change. The results of the graph are collected from DL.

B. Sociological Factors

Entity *S* sociological factors are summarized in Table II. Entity *S* has a total of six teams of sizes 5,5,4,5,3 and 15. As shown from the table, many team members are specialists with high level of domain and language(s) expertise. There is group of 3 testers who are shared between all the six teams.

C. Project-Specific factors

As shown previously in the sociological factors, teams in *S* work on large number of projects concurrently that makes it hard to collect project-specific factors for all of them. Table III collects some common factors for the projects on *S*. These factors indicate that the constraints affect all the projects performance and results.

D. Technological factors

The development methodology that *S* uses is the traditional Waterfall methodology. The programming languages used are all compatible with the web-based Oracle DB server. All project managers are PMP certified and they follow a rigid

Context factor	Value	Source
Team size (developers)	5 teams: Team 1 (5), Team 2 (5) Team 3 (4), Team 4 (5), Team 5 (3), Team 6 (15) + 3 shared testers (Total 40)	ER
Team Education level	Bachelors:35 Masters: 5 PhD: 0	ER
Experience level of team	1-5 yrs: 14,6-10 yrs: 22, 11-15 yrs:3, 16+ yrs:1	DL/IN
Domain expertise	High	DL
Language expertise	High	DL
Experience of Proj. Mgr.	High	DL
Specialist(s) available	IS Consultant,Dedicated DBA,Web services specialist	DL
Personnel turnover	5%	PT
Morale factors	Medium	DL

TABLE II
SOCIOLOGICAL FACTORS

Context factor	Value	Source
Domain	Web Applications	DL
Nature of projects	New, Enhancement or Maintenance	PT
Relative Complexity	Vary from Low to moderate based on the project type	PT/SC
Projects age	Some projects age back to 12 years	DL
Constraints	Forever-maintenance phase, No dedicated tester, No change management rules, Quality management process	DL

TABLE III
PROJECT SPECIFIC FACTORS

management style. A summary of the Technological factors is shown in table IV.

E. Ergonomic factors

Table V summarized the ergonomic factors of entity S. From the observation of the physical layout and office space, the open labs leads to high level of distraction that affects

Context factor	Value	Source
Soft. Eng. Methodology	Waterfall traditional methodology	DL
Project Management	PMP	DL
Defect Prevention and removal	Reviews and unit testing	PT/OB
External/system test	Done by testers after the end of the development phase.	PT/OB
Language	PL/SQL, Java, .NET	DL/PT
Reusable materials	Common design templates, Third party libraries, code template skeletons	DL/SC

TABLE IV
TECHNOLOGICAL FACTORS

Context factor	Value	Source
Physical layout	Open labs/workstation areas	OB
Distraction of office space	High	OB/IN
Customer communication	Customer contact person, face-to-face, emails, official letter communication.	DL/IN

TABLE V
ERGONOMIC FACTORS

Context factor	Value	Source
Team Location	Co-located	DL/OB
Customer cardinality and location	Remote (in different buildings or cities)	DL

TABLE VI
GEOGRAPHICAL FACTORS

the working outcome in general. There are many means of communicating with customers but the most official way is by formal letter, which leads sometimes to delays in project follow up.

F. Geographical factors

The last context factor in S is geographical factors. As shown in table VI, team and customer locations are relatively close, which spares the issues that come with multi-national and different time zones communication.

III. CHALLENGES OF AGILE ADOPTION IN ENTITY S

The agile teams in S have faced many issues during the Scrum adoption phase. The challenges were significant enough to put the Agile project on hold, teams started gradually to go back to the traditional Waterfall model. To understand such challenges, we conducted meetings and surveys with every development team and with the upper management of S. Many challenges were recorder, in this paper we list 8 challenges in priority order.

1) *Challenge 1: Missing the Agile Master Role:* Agile master or Agile coach is an essential role during agile adopting process in any organization [13], [4], [1]. Agile coach is considered a consultant for the team in every step of a project using an agile method, such as Scrum, that is responsible of providing guidance and help to succeed in adopting agile. Entity S management recognized the need to fill this new management role introduced by Scrum. However the position was not filled due to financial constraints and senior software managers from within entity S took the role as the Scrum Master.

2) *Challenge 2: The overzealous teams:* The Agile course that the employees of S attended had a very positive reaction and convinced the management about the benefits of the transition to agile practices. It created a number of overzealous teams that wished to adopt to agile as soon as possible hoping it will solve all their previous development challenges known for traditional methods. Only few members of the teams raised

concerns on fast adoption of a new method, but finally it was decided to go with the vote of the majority.

The overzealous team fast adoption of agile led to many unexpected problems. One of these problems is the decrease in productivity because the development cycle took longer time due to many mistakes in implementation. This decrease in productivity led many team members to be less optimistic and started to lose interest in agile methods. To catch up with deadlines the team started to go back to the previous waterfall method.

3) *Challenge 3: The Absence of a Pilot Project:* Another challenge is the absence of a pilot project in the transition from the previous traditional method to the scrum method [4]. Conducting a pilot project was a recommended step in the adoption of agile development for the first time and was advised by the *Introduction to Agile Development* training course.

As a part of the plan to adopt agile method, the pilot project is essential to evaluate how *S* environment will be able to move from the previous heavy-weight method to a new light method that is completely unfamiliar to the teams. In literature many organizations went through the same experience of running a pilot project especially those companies that has large projects in comparison with *S* such as Microsoft [14], Yahoo [15], [16], Intel [17] and Amazon [18]. After investing the needed time and resources they have reached to a successful adoption of agile.

After the management decision to adopt Scrum, there was a debate on the need for a pilot project. A meeting was held with the management to discuss the advantages of starting a pilot project as a first step before full adoption. The outcome was not very encouraging and the theme of the meeting can be summarized by a comment from one senior manager: *Why investing too much and wasting time and resources for a pilot project if the Scrum method promises success?* Later after few weeks of direct adoption of Scrum, the team started to face tremendous difficulties and started gradually going back to their previous work method to catch up with late deadlines.

4) *Challenge 4: Scrum Implementation:* Although the employees in *S* were very experienced but yet none of them had any previous experience with agile development methods or Scrum implementation in particular. This is in addition of the absence of the Agile master.

For the team members, scrum implementation was not as easy as it appeared to be during the training session. The team members found themselves, suddenly, in a completely new setup. The experience of traditional methods is completely different than committing to daily meeting, working with time boxes, finishing tasks in small period iteration and documenting the stories (or backlogs). Changing project management and programming habits in weeks or even months is difficult because it comes with training and more serious commitment to change.

5) *Challenge 5: Current Work Pressure:* Although *S* software development teams serve a very large organization of over 30 departments and developed numerous project through

the years. The development projects require continuous maintenance and support. In addition, the team was working on a new project with firm deadlines. The work environment was very demanding and team worked under pressure to produce products according to the planned schedule.

Scrum adoption process started while every member of the team was engaged in his/her everyday tasks. With such work pressure the daily Scrum meetings added extra pressure to the employees. They used to meet weekly and later twice a month and then only when required and usually after working hours. As teams started to skip daily meetings it also affected the learning process of scrum between the team members. This eventually led to the failure of learn and implement concept of the agile method.

6) *Challenge 6: Upper Management Concerns :* The upper management of *S* had many concerns about the effectiveness and success of the transition to a new method. The upper management was not easily convinced to invest in a new method.

Not having the full support of the upper management was a great challenge. The employees were unmotivated by lack of resources and lack of recognition of their efforts to improve the software development method. With time, and with the other challenges they faced, Scrum method implementation reached to an end.

7) *Challenge 7: Governmental bureaucratic System:* The traditional method currently in *S* was customized to comply with the governmental system of other department. The new agile method being introduced, Scrum, is developed in such highly bureaucratic environment. The agile team has to secure approvals and signatures before moving from one step to another. This was perceived by the team members as unnecessary and more time was taken into account to develop a new project. The scrum method requires much less correspondence, less time in communication between customer and the team and requires significantly less paper work and approvals as the customer is supposed to be involved in every step. Here, the governmental bureaucratic system came on the way of succeeding in the adoption of agile.

8) *Challenge 8: Documentation requirements:* After years and years of extensive documentation of every step in the traditional method, moving to a new method with minimum documentation requirements was one of the greatest challenges. Every project used to end up with dozens of documents such as project charter, project plan, testing plan, SRS, STS, technical documents, user manual, etc. Each of these documents contained large number of pages written by every member of the team and consumed hours of the valuable development time.

The documentation requirements were driven basically from the previous challenge (the governmental system), upper management, ISO certificate requirements and the traditional development method that is currently used. Although agile development promises sufficient documentation of the projects, it did not seem very convincing to the upper management when they end up receiving few documents in comparison with the

previous model of documentation.

Many attempts were made to try to balance between the upper management requirement regarding documentation and between adopting Scrum method. Agile teams started to increase the number of documents required for documentation and started to customize Scrum as much as possible to conform with all the upper management requirements of documentation norms. This did not work very well and it created extra burden on the agile teams.

IV. PROPOSED SUGGESTIONS TO OVERCOME THE CHALLENGES

In this section we summarize suggestions to overcome the above agile adoption challenges. These proposed suggestions are either best practices in agile development, solutions that are suggested by previous researches or our own suggestions.

A. Challenge 1 Solution:

In agile software engineering literature [18], [13], [11] the Agile Master has a critical role to play to ensure successful outcomes. Obviously entity **S** needs to hire an Agile Master for at least six months to increase the chance of succeeding in the agile adopting process.

B. Challenge 2 Solution:

Careful planning and discipline of agile adoption is required. A team must also expect a slight decrease in productivity when first adopting agile while the teams learn the new implementation techniques.

C. Challenge 3 Solution:

There is no doubt in the agile software community that the pilot project is an essential part of the adoption steps. A pilot project must be selected when first adopting agile. The pilot project should be carefully selected as follows [18], [12]:

- The project duration should be near the middle of what is the average for an organization
- It should be small enough to be done by one team.
- It should not be critical to the organization

D. Challenge 4 Solution:

Applying Scrum software development approach in an organization cannot be simply learned from a book or a course. To increase the chance of succeeding in scrum implementation several actions are advised:

- The agile master role must be provided to guide teams during the first 6 months of agile implementation
- Contracting or hiring an experienced developer with scrum might help and support the learning process of scrum between the employees.
- Increase the number of agile courses training and workshop that provide more details of how to implement scrum.

E. Challenge 5 Solution:

The adoption of agile should be planned in a time that has the minimum work pressure, for instance, entity **S** should invest more time by not accepting new projects for 6 months and investing this time only for agile adoption. Such an investment will pay back later after the teams get their appropriate training on scrum and start working efficiently with the new method.

F. Challenge 6 Solution:

Upper management approval is fundamental to support any major change in the processes of any department, without their approval the success of the adoption is doubted. To acquire the upper management support a presentation or a meeting might be conducted to demonstrate the new agile development and explain the benefits of changing to a new method.

G. Challenge 7 Solution:

One of the valuable benefits of agile development is its ability to be customized based on the culture and the environment of the organization it is adopted in. These changes should not compromise its beneficiary and affect its efficiency. It might not be realistic to change the whole organization system overnight but that does not mean that governmental systems should not be revised and enhanced to what is best for the governmental organization.

H. Challenge 8 Solution:

Lots of organizations complain about the lack of documentation, but also over-documenting and spending plenty of time documenting unnecessary information is incorrect. What agile development offers is a new way of comprehensive documentation that takes less time and effort. This is another issue that needs to be supported by the upper management and agreed upon with the customers from the beginning of the project.

V. CONCLUSION

Departure from plan-based software engineering approach to Agile software engineering approach is a major step that needs careful planning and requires allocation of resources. In this paper we present a case study on Agile adoption for an organization in the United Arab Emirates (U.A.E). The organization follows the traditional Waterfall software development model. Many challenges were faced by the agile teams which caused the adoption to fail. The challenges include: missing the Agile master, the overzealous teams, absence of a pilot project, unexpected difficulty in Scrum implementation, high work pressure, and many others. All these challenges could be avoided or minimized if the adoption process were carefully planned and enough human and financial resources were secured in advance [19]. Many lessons may be learned from such experience and it is worth sharing this case study with the industrial and research community

REFERENCES

- [1] G. Beneffeld, Rolling out agile in a large enterprise, *f* in *41st Hawaii International Conference on System Science*. 2008, IEEE Computer Society.
- [2] O. Salo and P. Abrahamsson, European embedded development organizations: a survey study of extreme programming and scrum, *f IET Software*, vol. 2, pp. 58~64, 2008.
- [3] Richard Moore, Kelly Reff, James Graham, and Brian Hackerson, Scrum at a fortune 500 manufacturing company, *f* in *Proceedings of the AGILE 2007*, Washington, DC, USA, 2007, pp. 175~180, IEEE Computer Society.
- [4] M. Cohan, *Succeeding with Agile Software development using scrum*, Addison-Wesley, 2009.
- [5] Kent Beck, Mike Beedle, Arie van Bennekum, Alistair Cockburn, Ward Cunningham, Martin Fowler, James Grenning, Jim Highsmith, Andrew Hunt, Ron Jeffries, Jon Kern, Brian Marick, Robert C. Martin, Steve Mellor, Ken Schwaber, Jeff Sutherland, and Dave Thomas, Manifesto for agile software development, *f* 2001.
- [6] V. Basili, F. Shull, and F. Lanubile, Building knowledge through families of experiments, *f IEEE Transactions on Software Engineering*, vol. 25, pp. 456~473, 1999.
- [7] R.k.Yin, *Case Study Research: Design and Method*, Sage Publications, 3rd edition, 2002.
- [8] B.A. Kitchenham, L. Pickard, and S.L. Pøeeger, Case studies for method and tool evaluation, *f IEEE Software*, vol. 12, no. 4, pp. 52~56, Jul 1995.
- [9] B.A. Kitchenham, S.L. Pøeeger, L.M Pickard, P.W Jones, D.C. Hoaglin, and J. Rosenberg K. El Emam, Preliminary guidelines for empirical research in software engineering, *f IEEE Transactions on Software Engineering*, vol. 28, no. 8, pp. 721~734, Aug 2002.
- [10] Tore Dybbø and Torgeir Dingsøy, Empirical studies of agile software development: A systematic review, *f Information and Software Technology*, vol. 50, pp. 833~859, August 2008.
- [11] B. Boehm and R. Turner, *Balancing Agility and Discipline: A guide for the Perplexed*, Addison-Wesley, Boston, MA, 2003.
- [12] C. Jones, *Software Assessment, Benchmarks, and Best Practices*, Addison-Wesley, Boston, MA, May 2000.
- [13] J. Tabaka, 11 ways agile adoption fails, *f StickyMinds.com Column*, stickyMinds.com., retrieved March 2011.
- [14] Andrew Begel and Nachiappan Nagappan, Usage and perceptions of agile software development in an industrial context: An exploratory study, *f* in *Proceedings of the First International Symposium on Empirical Software Engineering and Measurement*, Washington, DC, USA, 2007, ESEM '07, pp. 255~264, IEEE Computer Society.
- [15] K. Scotland and A. Boutin, Integrating scrum with the process framework at yahoo! europe, *f* in *Agile 2008 conference*, Toronto, ON, 2008, pp. 191 ~ 195.
- [16] G. Cloke, Get your agile freak on! agile adoption at yahoo! music, *f* in *Proceedings of the AGILE 2007*, Washington, DC, USA, 2007, pp. 240~248, IEEE Computer Society.
- [17] B. Greene, Agile methods applied to embedded firmware development, *f* in *Proceedings of the Agile Development Conference*, Washington, DC, USA, 2004, pp. 71~77, IEEE Computer Society.
- [18] M. Cohan, Tips to select a pilot project for agile adoption, *f* from Infoq: <http://www.infoq.com/news/2009/11/pilot-project-agile-adoption>, November 2010.
- [19] Jeffrey A. Livermore, Factors that significantly impact the implementation of an agile software development methodology, *f Journal of Software*, vol. 3, pp. 31~36, April 2008.