International Journal of Engineering & Technology, 7 (4.31) (2018) 198-203

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Enhancing Scrum Framework: a Case at a Multinational Manufacturing Company in Malaysia

Shalini Mariyappan[1], Nasuha Lee Abdullah[2]*, Rosnah Idrus[3]

School of Computer Sciences, Universiti Sains Malaysia
**Corresponding author E-mail: nasuha@usm.my*

Abstract

Many organizations today are adopting Scrum, an Agile methodology for handling complex software project. Despite all the benefits, there are still issues and challenges in adapting to Agile and some refusals to move from traditional method. This research aims to assess the perception of effectiveness of implementation of Scrum and identify the challenges after adopting Scrum in an information technology (IT) department of company X, a multinational manufacturing company in Malaysia. The findings of this study revealed that after one year of adopting Scrum, the perceptions from the staff is there is no significant improvement made. Five challenges were identified: team members unaware of each other's tasks, difficulty in handling daily task priorities, no work synchronization among management and delivery team, too many tasks at the middle of sprints and poor attendance during daily stand up meeting. An enhanced Scrum framework that uses Scrum and Kanban is proposed to address the identified challenges. The insights gained from this research hope to provide practitioners with useful reference for adopting scrum.

Keywords: *Agile; Challenges; Framework; Kanban; Scrum*

1. Introduction

Agile software development has emerged into the software community for a decade now and is replacing the traditional software development [1]. This wide-spread adoption is due to its ability to respond to fast changing business needs, technologies and market conditions as it emphasize mainly on collaboration, people-oriented approach to software development. Agile software development has been recognized for following a set of rules such as working software over detail documentations, people and interaction over tools and processes, collaborate with customer over official arrangements, and responding to change over following of the plan [2].

Scrum is a subset of Agile development method for completing complex projects which emerged in 1990s [3]. It is an Agile project management framework for software development projects to deliver new software capability every two to four weeks using iterative and incremental practices. The term Scrum originated from popular sport, Rugby in which there are 15 players each in two teams competing with each other. The strategies adopted from Rugby and incorporated into Scrum are holistic team approach, persistent interaction among team members, and unchanging core team members [3]. Scrum is widely used by software development teams and after some time it has spread to other business functions such as IT and marketing. Adopting pure Scrum process has many benefits to the organizations that includes higher productivity, reduced time to market, increased job satisfaction, lower bug rates, quicker adaptation to rapidly changing business needs and many more [4]. The meaning of pure Scrum is that the framework is free from extended and costumed version of the Scrum model. Despite of all the benefits, some still find it a challenge in implementing pure Scrum especially in organizations that could not adapt to regulated environment.

Company X is a multinational manufacturing company producing semiconductor devices based in Penang, Malaysia. One of the largest departments in company X is the IT department that has implemented pure Scrum framework in their operations. Observations have found that employees are experiencing stress and have difficulties to adapt. Company X implemented Scrum to be more productive and efficient. However, after implementing pure Scrum for a period of one year, it still failed to provide the desired outcome. Hence, the objectives of this research are 1) To assess the perception of effectiveness of Scrum implementation 2) To identify challenges encountered by the employees after adopting Scrum 3) To propose an enhanced Scrum framework which can help both organization and employees to appreciate the benefits from implementation of Scrum. At the end of this study, an enhanced Scrum framework that combined Scrum with suitable features from Kanban, another subset of Agile methodology, is proposed to address the identified challenges in company X. In addition, an increasing number of software organizations stressed that tailoring an Agile methodology according to company needs is important [4].

2. Related Works

This section provides an overview of Scrum and Kanban. Features and challenges associated with Scrum and the benefits of Kanban are discussed. The section ends with a comparison between the two frameworks, which helps to show the benefits of implementing either Kanban or Scrum.

2.1. Scrum

Scrum is a well-recognized iterative and incremental technique resulting in working software to be distributed to customer at the end of each development iteration, known as sprint. It is focused on dealing with software development projects through firmly defined roles, meetings and process artifacts [5]. Scrum's structure supports empirical process control for managing projects. Scrum uses time-boxed iterations contrasting traditional methodologies where the degree of a delivery is acknowledged by the time needed. This means that instead of enabling the scope to create a release length, it defines the scope in a limited time frame and moderates the project risks [4]. In iterative models, all steps are repeated through until the system or task is deemed complete, rather than visiting each step only once. Projects are separated into small work cadences, known as sprints. The duration of each sprint is between one to four weeks. A sprint itself comprises of planning, development, integration, testing, and delivery. Usually customers and developers will actively participate in the process of development and dynamically rank features during sprint planning meeting which will be held in the commencement of each iteration. A working system is delivered at the end of each sprint. Throughout the sprint, there will be daily standup meetings for 15 to 30 minutes to allow participants constantly to control the authenticities especially for projects with lots of changes. As part of a self-organizing team, team members organize their work in daily standup meetings. To foster an environment of learning and adaptation, stakeholders together with developers and end users go through frequent cycles of thought-action-reflection during the conclusion of each iteration. These sprint review meetings help to gather feedback and reflection on what went well, what did not go well and key learnings. Sprint retrospective planning is to examine the development team itself and plan for enhancements to be endorsed during the next sprints [6].

The Scrum Framework [7] is as shown in Figure 1.

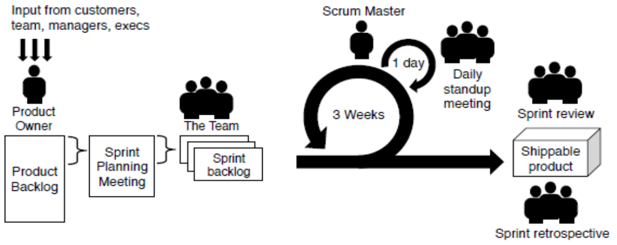

Fig.1: Scrum Framework

Scrum framework consists of product backlog, sprint planning meeting, sprint backlog, sprint retrospective, sprint review and daily standup meeting which will be followed by the team members for the whole completion of a project. Product backlog consist of all the tasks need to be done within the project. It contains the list of all structures, functions, requirements, enrichments and fixes that create the changes to be made to the product in upcoming releases. The Product Backlog items consist of the attributes of an explanation, order, estimate, and value. These items are normally labelled as User Stories. The Product Backlog, including its content, availability, and ordering is the responsibility of product owner [8]. The Sprint Backlog consist of all the committed User Stories for the existing Sprint broken down into tasks by the team. To accomplish the commitment, all items on the Sprint Backlog should be developed, verified, documented and integrated [9]. Finally, sprint retrospective is a meeting hosted by the Scrum Master where the team reflect on their achievement and lessons learned so that improvement can be made in next sprint. The sprint retrospective is a vital mechanism which helps a team to make continuous improvement in the project lifecycle [9].

2.2. Challenges in Scrum

There were very few research conducted to identify challenges in Scrum as most researches support and conclude that the implementation of Scrum brings benefit to the organization. However, recently [10] have conducted a case study in two software maintenance teams and identified few challenges in Scrum that has led the team to migrate to Kanban. The challenges identified are 1) lack of work visibility 2) fluctuating task priorities 3) over commitment of sprints 4) lack of communication and collaboration and 5) lack of work synchronization. According to [4], lack of work visibility was identified as a major issue as team members were unaware of each other's task. Not only it is vital for successful completion of tasks but also the right level of information needed to be visible to the management at the right time. In order to deliver value to customers and reduce risk of costly software failures, task prioritization is an activity that needs to be performed carefully. In the case reported by [4], fluctuating task priorities or constant reprioritization of tasks made it difficult to complete the work on time and made it challenging to find an equilibrium between development and maintenance work. The worst scenario is when the teams have to abandon their current work to deal with emergency or high-priority tasks [10]. Fluctuating task priorities may lead to over commitment of sprints. In Scrum, team commit to ensure a certain amount of work is done at the end of each sprint. Even though Scrum protects the team from unexpected work in each sprint and delivers the committed work, it cannot prevent intervention from higher management. For example, in the middle of a sprint, team members need to accept more tasks assigned from higher management and causing the team unable to achieve its sprint commitments [10]. Communication posed quite a challenge especially when many teams are working together concurrently on a project. In Scrum, a daily standup meeting is to foster regular communication and provide latest update on team members and project work. It became a challenge when some of the team members do not attend the meeting regularly or not paying attention during the meetings [10]. This causes lack of communication and collaboration and made teams work in isolation. Management needs up-to-date information about the progress of any project, which is important for resource allocation and task distribution. Based on the study, it was found that the lack of work synchronization with management makes it very difficult to change team members from one project to another. The lack of work synchronization is also a symptom of poor coordination inside and outside the teams as well as with management [10].

2.3. Kanban

Kanban is derived from a Japanese word, which literally means 'signboard' and it is an actively used in manufacturing industry as a scheduling system that uses pull concept known as 'just in time' [11]. In 2004, David J Anderson introduced the use of Kanban in a software development project to improve the performance of an IT team in Microsoft. The use of Kanban has improved the visualization of the flow of project, reduced work in progress (WIP) and enable measurement of cycle time [11]. A maintainable pace of development can be attained, better quality products delivered and increased team performance by restricting work in progress. The grouping of enhanced flow and improvised quality software helps to abbreviate lead time, leading to regular releases that helps to build strong relationship with the customers. The Kanban board delivers visibility to the software process, because it displays task assigned for each team member, clearly indicates priorities and highlights holdups. Additionally, its goal is to diminish work in progress where it requires to develop only those items which are requested. As the developers focus only on those few items at a given time, this enables endless flow of released work items to the customers. The goal of Kanban is to use shorter feedbacks loops to quickly familiarize the process. The key incentive for the usage of Kanban is emphasis on flow and the absence of mandatory iterations. Since there is a lot of advantages achieved in manufacturing industry by practicing Kanban, this method is expected to be highly effective in software development as well. All the IT professionals are convinced to adapt Kanban based on successful histories in manufacturing field [12]. An example of a physical Kanban board is as shown in Figure 2.

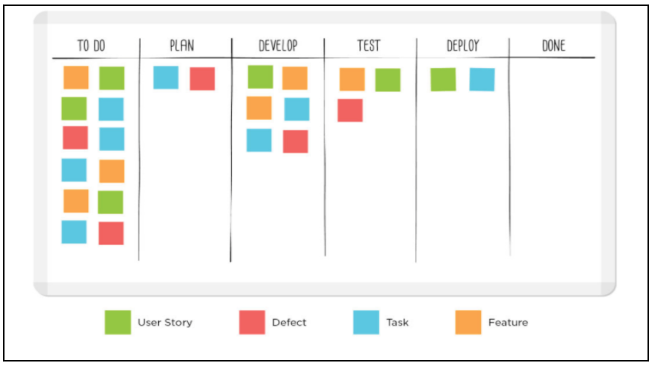

Fig. 2: Physical Kanban Board

Kanban became prevalent due to easy execution, useful visual control, manageable work in progress, and focus on the continuous process improvement. Kanban follows a simple mechanism whereby the work in progress (WIP) is restricted and new task will only be initiated when the current task is completed. It is also known as "pull" system whereby instead of pushing a new task into the system, it will be dragged into the system when there is a capacity to handle it. WIP limit the capacity of each step in the development process to ensure not to overload a pull system. It helps to quickly bring out issues that may affect performance. WIP limit helps to highlight the current constraint on the system and prompt the team to resolve the issues before continuing with next tasks [13].

Kanban board has user stories presented and it is the main requirement of Kanban, which is classically contains sticky notes on a white board or an electronic card wall system which assists as a visual control mechanism representing how the work flows through the numerous stage of development process. Kanban board can visualize various state of a work item that can occur during the development process in a sequence of columns. The cards travel from one column to another until they end in the last column as work progresses through the development lifecycle. Each column has a WIP limit that specify the number of cards allowed to be in the equivalent workflow state at any one time. When a task (card) is completed, it travels to the next column leaving a space in its existing column, this allows the team to pull a completed card from a preceding column. If there is an instance where the tasks (cards) on one column cannot be completed, it will eventually hit its WIP limit and prompts the team to resolve the constraint before moving on to next tasks [13]. The visibility provided by Kanban helps to promote team work among all the stakeholders to strive for continuous improvement. Kanban can be easily added into any development process that needs monitoring and team work [13].

2.4. Comparison of Scrum and Kanban

According to [4] both Kanban and Scrum belongs to Agile and Lean methodology where the focus is to reduce time to market. In their study, it was concluded that Scrum has more constraint and leaves open fewer options. Scrum is based on time-boxed iterations while Kanban is not. The researcher concluded that Kanban is more suitable for work in which there is a high variability in priority. However, it has been also justified that Scrum is more suitable for work with tasks that can be prioritized in batches. Based on the case study presented by the researchers, Kanban offers numerous benefits to maintenance work, such as bringing visibility to maintenance tasks, protecting teams from over-commitment, helping in task prioritization, synchronizing work with other teams and management, easily changing team members between teams, improving communication and endorsing collaboration in teamwork. Table 1 shows the comparison between Kanban and Scrum.

Table 1: Differences between Scrum and Kanban method

Kanban	**Scrum**
There are no prescribed roles	Predefined role for Scrum master, product owner and team member.
Continuous delivery	Time-boxed sprints
Work is extracted in a single piece flow.	Work is extracted in batches.
Changes in task can be made any time	No changes allowed in the middle of a sprint
Default metric for planning and process improvement is the lead time	Default metric for planning and process improvement is the velocity
Suitable in operational environments with priority in a high degree of inconsistency	Suitable in situations where work can be prioritized in batches that can be left alone
Optional commitment	Team commits to a specific amount of work
WIP restricts directly	WIP restricts indirectly
A Kanban board is determined	A Scrum board is retuned between each sprint

Based on the review of literature, there are multiple Agile software development frameworks apart from Scrum. The frameworks are: crystal methodology, dynamic software development, feature-driven development, lean software development, extreme programming [14]. However, there is no any framework which is more superior than the other and in fact every framework has its own challenges. Finally, to ensure sustainable and effective implementation of Scrum, it is essentials to take into considerations the challenges highlighted from the literature together with the inputs from practitioners.

3. Research Methodology

The research method in the study adopted the Design Science research process model. The main reason for adopting this method is the emphasis of 'problem-solving / performance-improving' nature of the activity which suits the core idea of this study. In fact, this method is sometimes known as 'improvement research' [15]. The model is as shown in Figure 3.

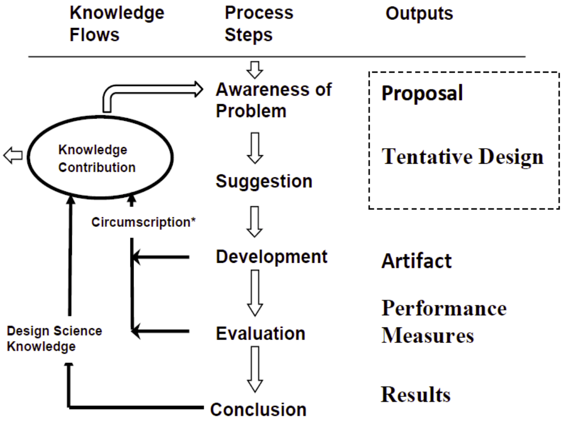

Fig. 3: Design Science research process model [15]

The research process begins with awareness of problem, suggestion, development, evaluation of the development and finally conclusion [15]. Each of the steps is described as follows: -

Awareness of problem: An awareness of the problem raised from the new development in the industry in which the transition from traditional waterfall method to Agile methodology. Observation conducted found that employees were experiencing stress and have difficulties to adapt to Scrum. The output of this step is a proposal to enhance the current Scrum implementation in company X. A questionnaire was developed and distributed to the Scrum teams to solicit the challenges faced by the Scrum teams, their perceptions on the effectiveness of the implementation and suggestions for improvement. There was a total of 100 employees in the IT department and 60 employees responded to the questionnaire. The questionnaire was distributed using WhatsApp group and emails. It took around one week to gather all the responses.

Suggestion: After analyzing the results from the questionnaire, a tentative design was suggested. Suggestion is fundamentally a creative step whereby new functionality is proposed based on the existing Scrum model. An enhanced Scrum framework is suggested.

Development: The tentative design is further developed and implemented in this phase. The enhancement incorporated in the new design can resolve all the challenges identified.

Evaluation: Once constructed, the proposed enhanced framework is evaluated based on the cycle time. This metric helps to provide an overview of the team performance and evaluate effectiveness after implementation. However, for the purpose of this paper this step is not included as the study is still ongoing.

Conclusion. This step concludes the research cycle. The outcome is whether the results have achieved objectives of the research.

There are three parts in the questionnaire. Part A is the demographic of the respondents. Since this study is intended for Scrum users only, the role of the respondent in Scrum team is useful in understanding the relationship between their roles and their feedback on Scrum. For example, product owner may have different perception about Scrum as they work more closely with the higher management while Scrum master and team members might have different views as they are working in the real environment and might encounter a lot of challenges that are not visible to the management. The second part, Part B is to gauge the perceptions on effectiveness of the implementation of Scrum in company X. Ordinal scales of 1 to 5 were used to measure the perception on effectiveness, quality of work, team performance as well as satisfaction level of adopting Scrum. Scales of 1 indicates strongly disagree, 2 disagree, 3 neutral, 4 agree and finally 5 indicates strongly agree. The final part, Part C of the questionnaire was opinion seeking on the challenges and recommendations for improvement. Respondents were asked to select as many challenges from a list of nine issues. These challenges were obtained during the observation done at the problem awareness step as well as those found by [4]. Respondent was allowed to add new challenges that were not in the list too. The nine challenges are: 1) team members are unaware of each other's task 2) difficult to handle daily task priorities 3) no work synchronization among management and delivery team 4) too many tasks at the middle of sprint 5) poor attendance in the daily stand up meeting 6) overall task status are missing when task are on block status 7) cannot achieve time-box goal 8) difficult to build trust with customers 9) loss of management control.

4. Results and Discussions

This section discusses the results obtained from the questionnaire. The data is analyzed using quantitative analysis. As stated above, 60 responses were collected from those who involved directly in Scrum at company X. The results from the analysis are reported according to part A, B and C from the questionnaire.

4.1. Demographic Analysis – Part A

The respondents consisted of 31 female and 29 male with 73% of them having more than 5 years of service. This indicated that the respondents were experience workers. 50% of the respondents assumed the role as team members, 20% of them were product owner and 30% were Scrum Master. This indicated that the responses collected covers all the roles in a Scrum team.

4.2. Perception of Scrum Implementation – Part B

The results on perception on effectiveness, quality of work, team performance as well as satisfaction level after adopting Scrum is as shown in Figure 4. The responses were categorized based on Negative perception (scale 1, 2), Neutral (scale 3) and Positive perception (scale 4, 5).

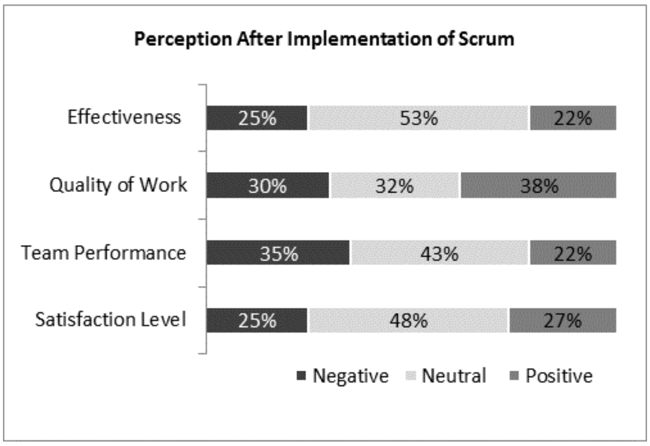

Fig.4: Perception after Implementation of Scrum

Generally, the results were not very conclusive as majority of the responses were neutral especially for effectiveness and level of satisfaction. This result imply that the employees were somewhat made to implement Scrum and have yet to reap the benefits from it.

4.3. Challenges and Recommendations – Part C

A total of 245 responses were recorded in Part C of the questionnaire since the respondents were allowed to select as many challenges from the list of nine issues. Two new issues were highlighted, no prior knowledge on Scrum and unsure of Scrum process and methods. Out of the 245 responses, the top five challenges are 1) team members are unaware of each other's task, 43 responses 2) difficult to handle daily task priorities, 34 responses 3) no work synchronization among management and delivery team, 34 responses 4) too many task at the middle of sprint, 32 responses 5) poor attendance in the daily stand up meeting, 32 responses. All the top five challenges found in this study are similar with the issues highlighted by [4]. Meanwhile, several recommendations were obtained from the responses, which include call to incorporate features from Kanban. Among the features are 1) complete one task before accepting another task 2) provide visibility on bottlenecks 3) remove time-box iterations 4) visualize task in every workflow state 5) use lead time as metric for planning and process improvement.

These results imply that implementation of Scrum was not perceived to be effective and there are still challenges encountered by the employees in the current Scrum framework. However, based on the recommendations obtained from the study, these challenges can be resolved by adopting some features from Kanban. Hence an enhanced Scrum framework is proposed.

5. Proposed Implementation

The solution proposed is to introduce an enhanced Scrum framework, which incorporates features from Kanban. The proposed framework is as shown in Figure 5 where a Kanban board is added to the existing framework.

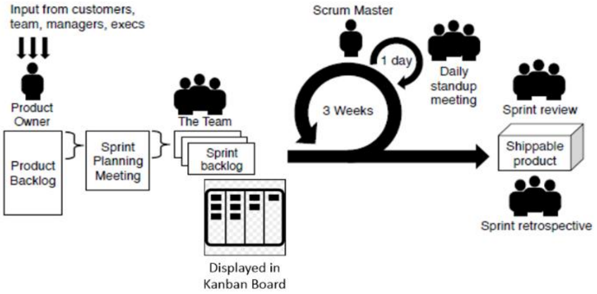

Fig. 5: Proposed Enhanced Scrum framework

Table 2 shows how the features from Kanban board could overcome the challenges identified.

Table 2: Challenges and Proposed Solutions

Challenges	Proposed Solutions
Team members are unaware of each other's task	Kanban board helps to visualize the work of each team member's progress in each phase. Unlike Scrum board that needs to be reset between each iteration. Kanban board no need reset and team members will be aware of their individual tasks and their tasks are visible to outside of their team too.
Difficult to handle daily task priorities	The team can now competently handle frequent reprioritization and constantly changing task using Kanban board as Kanban works based on highest priority task. The team becomes highly responsive without overheads as Kanban is based on continuous delivery model compared to Scrum that is based on time-boxes.
There is no work synchronization among management and delivery team	Improved task visibility using Kanban board enable the management to be more aware of individual team member's work. It helps management in resource allocation and synchronization especially tasks in blocked status which requires assistance from other team.
Team started to work on too many tasks at the middle of sprint	Kanban encourages team to work in continuous manner by applying WIP limits and practicing pull technique to work on highest priority tasks rather than involving in unscheduled tasks.
Poor attendance during the daily stand up meeting	Kanban has daily stand up meeting similar in Scrum. However, it is more effective as team members need to update their status in Kanban board before attending the meeting. During the meeting, member of the team can use it as an opportunity to seek assistance especially when their tasks are on blocked status. Team members can also offer to help other members if their assigned tasks are on hold.

Apart from overcoming the challenges, the Kanban's features are also meeting the recommendations from the respondents in the study. The proposed implementation would result in the changes summarized in Table 3. The comparison of the differences before and after (would be) implementation.

Table 3: Pure Scrum versus Enhanced Scrum

Features	Before (Pure Scrum)	After (Enhanced Scrum)
Ceremonies	Daily scrum meeting, Sprint planning, retrospective and sprint review.	Daily standup meeting, review and retrospective meeting.
Iterations	Using sprints	Continuous flow
Team members	Cross-functional	Can be within same functional and specialized
Team Roles	Product owner, Scrum Master and team members	Retained the same roles as Scrum team
WIP	Controlled in sprints which is 2 to 4 weeks	Using workflow state
Requirement Changes	Wait for next sprints	Added as needed in Kanban board

The proposed enhanced Scrum framework not only could resolve the identified challenges but most importantly to help the employees to appreciate the benefits of implementing Scrum.

6. Conclusion

This study hoped to help the employees in IT department of Company X to appreciate and reap the benefits with the implementation of Scrum. The findings indicate that the employees are still trying hard to adapt to Scrum even though it has been implemented for one year in the department. The findings obtained from this study are consistent with the findings by [4] that it is more effective to tailor or customize Agile methodology according to the needs of each organization. Limitation of this study is that the work is still ongoing, and evaluation is yet to be carried out for the implementation of enhanced Scrum framework. Nevertheless, the insights obtained from this study hope to fill the gap in empirical knowledge related to implementation of Agile methodology.

Acknowledgment

The authors would like to acknowledge the support from Universiti Sains Malaysia for the publication of this paper through USM Short Term Grant 304/PKOMP/6313219.

References

[1] Ktata, O. and Lévesque, G. (2009), "Agile Development: Issues And Avenues Requiring A Substantial Enhancement Of The Business Perspective In Large Projects", Proceedings of The 2nd Canadian Conference on Computer Science And Software Engineering, ACM, pp.59-66.

[2] Rouse, M. (2011), "Agile Manifesto", Retrieved on 20 July 2017 from:http://searchcio.techtarget.com/definition/Agile-Manifesto.

[3] Bannink, S. (2014), "Challenges in the Transition from Waterfall to Scrum – A Case Study at Portbase", 20th Twente Student Conference on IT. Enschede, The Netherlands.

[4] Yilmaz, M., & O'Connor, R. (2016), "A Scrumban Integrated Gamification Approach to Guide Software Process Improvement: a Turkish case study", Tehnicki Vjesnik (Technical Gazette), Vol.23, No. 1, pp.237-245.

[5] Nikitina. N, Mattsson. M. K & Strale. M (2012). "From Scrum to Scrumban: A Case Study of a Process Transition", Proceedings of the International Conference On Software and System Process, IEEE press, pp. 140-149.

[6] Ozkan, N. (2015), "Risks, Challenges and Issues in a Possible Scrum and COBIT Marriage", Proceedings of Software Engineering Conference (APSEC), 2015 Asia-Pacific, IEEE press, pp. 111-118.

[7] Fitzgerald B, Stol KJ, O'Sullivan R, O' Brien D (2013), "Scaling Agile Methods to Regulated Environments: An Industry Case Study", Proceedings of 35th International Conference on Software Engineering (ICSE)2013, San Francisco: IEEE Publications, pp. 863-872.

[8] Todd. J (2014), "Product Backlog Item's estimates", Retrieved on 19 July 2017 from:https://www.scrum.org/forum/scrum-forum/5652/product-backlog-items-estimates

[9] Devendra. R (2014), "Key Elements of the Sprint Retrospective", Retrieved on 19 July 2017 from:https://owl.english.purdue.edu/owl/resource/560/10/

[10] Ahmad, M. O., Kuvaja, P., Oivo, M., & Markkula, J. (2016), "Transition of Software Maintenance Teams from Scrum to Kanban", Proceeding of 49th Hawaii International Conference on System Sciences (HICSS), IEEE, pp. 5427-5436.

[11] Kniberg, H, “Kanban vs Scrum – How to make the most of both”, Retrieved on 19 July 2017, via: www.crisp.se/file-uploads/Kanbanvs-Scrum.pdf.

[12] Ahmad, M. O., Markkula, J., & Oivo, M. (2013), “Kanban in software development: A systematic literature review”, Proceeding of 39th EUROMICRO Conference of Software Engineering and Advanced Applications (SEAA), 2013, IEEE, pp. 9-16.

[13] Mahnic, V. (2014), “Improving Software Development through Combination of Scrum and Kanban”, Recent Advances in Computer Engineering, Communications and Information Technology, Espanha.

[14] Dybå, T., & Dingsøyr, T. (2008), “Empirical Studies of Agile Software Development: A Systematic Review”, Information and Software Technology, Vol.50, No. 9, pp.833-859.

[15] Vaishnavi, V. K., & Kuechler, W. (2015), Design Science Research Methods and Patterns: Innovating Information and Communication Technology, CRC Press, pp.17.