

An Empirical Study of Scrumban Formation based on the Selection of Scrum and Kanban Practices

Mashal Alqudah[#], Rozilawati Razali[#]

[#] Center for Software Technology and Management, Faculty of Information Science and Technology

Universiti Kebangsaan Malaysia, 43600, Bangi, Selangor, Malaysia

E-mail: mashal_alqudah@yahoo.com, rozilawati@ukm.edu.my

Abstract – Scrumban is a combination of practices from Kanban and Scrum practices to manage the development of software based on different project situations. Scrumban practices formation can differ based on different projects and team members. However, since each method has its own pros and cons, inappropriate formation of Scrumban practices may lead to increased waste and time of development, and decreased quality, which in turn, affect the Agile organizations and cause inefficient and ineffective development. Practitioners of Kanban and Scrum are convinced that a combination of both methods is better than the use of one and thus, practitioners should be guided in their decision making. This study aims to show how Scrumban method is formed based on a combination of Kanban and Scrum methods. Scrumban formation and the identification of the factors, which assist in the combination of Kanban and Scrum were conducted through a review of the previous work and semi-structured interviews with 7 Agile experts, after which, content analysis was conducted to analyse the gathered data. Different factors - the method prescription, roles and responsibilities, adoption time, team size, batch size, requirements prioritization, feature size, lead time, technical practices, cost and quality, assist Agile team members in the formation of Scrumban by combining appropriate Kanban and Scrum practices. In addition, Scrumban were found to be more appropriate than Scrum or Kanban in saving time, improving quality and minimizing waste.

Keywords— scrumban; formation; selection factors; Kanban; scrum.

I. INTRODUCTION

Agile method is the underpinning conception to a set of software development methods. It corresponds to four values and twelve principles of the Agile Manifesto [1]. Scrumban method, which is a Hybrid of Scrum and Kanban methods, was invented by these values and principles. The values above focus on working software, individuals and interactions, customer collaboration, and reacting to change quickly [1]. The captured values are refined into 12 principles as shown in the alliance's manifesto [1], Several of the principles are as follows [1]:

- Stakeholders (users and customers) and developers must cooperatively work all through the project.
- Projects must be constructed around individuals who are driven.
- The direct discussion is the most effective and efficient method for information transmission within the team of development and delivery.

- Working software is the most appropriate method for measuring progress because it quantifies the value of the business.
- A constant pace should be maintained to enable sustainable delivery and development.

For the fulfillment and achievement of these values and the principles, a range of practices are presently available to be applied alongside Scrum and Kanban methods, as can be referred in Table I and Table II. Based on the outcomes of the 2013 IT Project Success Rates Survey, Agile methods appear to be more effective and are also more successful, in comparison to the structured methods [2]-[4].

As one of the first-generation Agile methods, Scrum has been the popular choice among firms that decide to adopt the Agile approach [7]-[14]. On the other hand, as a second-generation Agile method [3],[6], the Kanban method is commonly applied by organizations worldwide especially by those in Europe [15],[16]. The use of Scrum and Kanban methods can be seen in countless organizations worldwide [17]. Still, members of Agile team recognize the non-

existence of one method that caters to all [15],[18]-[20], neither Kanban nor Scrum is appropriate for all projects [15]. Hence, a hybridization of both methods is needed.

As indicated by Scrum Alliance, Scrum comprises a small assortment of practices widely utilized in the industry [8]. Scrum team functions as Product Owner (PO), a Scrum Master (SM), and a Development Team [21],[22]. In Scrum, each team accomplishes their tasks by four artifacts as follows: a Product Backlog, a Sprint Backlog, a Product Increment, and Done Definition [21]. Further, five tasks are to be accomplished by the Scrum team to attain its goals, and they are Refinement of the Backlog, Regular Meeting on a daily basis, Planning, Retrospective, and Reviews considering each sprint. Based on these, the method of Scrum is viewed as prescriptive [21].

Comparatively, the Kanban method is not as detailed concerning the practices and principles it requires. Kanban is not as rigid as Scrum [3],[5]. Nonetheless, due to its ability in handling the challenges which could not be tackled by past methods, Kanban is deemed as a potent method [6]. Kanban also shows better effectiveness for teams in the incremental execution of business value [6]. During Kanban implementation, the key practices for process evolution generally include workflow visualization using Kanban board, restriction of work in progress through the decrease in the number of features under doing list, measurement, and management of the flow, policies clarification, feedback, and constant and cooperative looping and improvement [23]. However, just like Scrum, Kanban also requires teams that are highly self-organized and high leadership involvement in generating other practices. This would facilitate the use of Agile within the organization. Also, the choice of Agile practices will impact the success of software development [24].

Therefore, the main aim of this study is to discern how to select the optimal practices from Scrum and Kanban to form Scrumban successfully. Also, the factors assisting the Agile team members in the successful merging of Scrum and Kanban to be used in the implementation of many projects are examined. In this regard, the disposition and limitations of the methods are taken into account in order to assure successful result.

Hence, firstly, the methods are reviewed to understand the practices, principles, similarities as well as differences between these methods. At the same time, the selection factors which help in Scrumban formation are taken into account. Furthermore, in-depth semi-structured interviews with Agile experts were conducted, with the aim to examine and understand the way. Agile team members select and combine both methods for various projects in several organizations.

The merging of diverse practices from different Agile methods takes into account diverse factors [25]. Further, management of the project is a concern to all members of the team in assuring successful software projects [26], and this applies to any organization. Relevantly, methods such as Kanban and Scrum prove their suitability in the successful development of projects and their management as well [27]. From the past works, it is clear that choosing either the Kanban or the Scrum method or both in combination (Scrumban) needs an understanding of the factors

influencing the selection or hybridization. Further, harmonizing the strengths of each method by the method direction value, time of adoption, responsibilities, and roles, size of the batch, feature, and team, requirements prioritization, time, practices, quality, and cost, is essential. This is because the harmony assists the team members in deciding if they should employ just one method or a hybrid method.[27].

II. MATERIAL AND METHOD

The present section highlights a brief history of Scrum, Kanban and Scrumban methods, particularly concerning their first emergence, their primary roles, practices, as well as definitions.

A. Scrum Method

Scrum method relates to an approach that allows the generation and maintenance of multifaceted products. This method is simple as well as lightweight, but its full comprehension and implementation are still challenging [21]. Nonetheless, Scrum allows the use of different processes and methods among team members [21]. Scrum adheres to the rules that link the events, roles, and artifacts while also controlling the relationships and interactions that occur among them. Here, the emphasis is on the self-organization of teams not motivated by outside members.

Additionally, the use of Scrum requires a team size that is sufficient in retaining its flexibility and yet sufficiently large to complete sizeable amount of work within a sprint. In this regard, the development team size is 5 to 11 members [21].

Scrum involves various roles [16],[28],[29], which makes it different from Kanban. As an example, product owner assumes the major accountability in managing product backlog, while the development team assumes the task of dispensing a releasable sprint, whereas Scrum master assumes the task of assuring that every related party understands the Scrum method and that the method is employed effectively [21][29]. Scrum master functions as a servant-leader of the development team members. It highlights the practices related to the Scrum method with each practice briefly defined.

B. Kanban Method

The method of Kanban by Anderson (2004) was created during the attempt to save a small Microsoft IT team [16]. Somehow, the author only exposed the Kanban's details to software development in 2010 [23],[28]. The main objective of Kanban is to protect the Agile team members from the interminable tasks. In this regard, the purpose is to recognize the constant speed of development as well as the change that is akin to the conventional Agile methods, where there is little to no opposition to change [28],[39]. In particular, the Kanban method is grounded on the premise of Just-in-Time (JIT), especially in software development. Hence, the team is not supported in developing or delivering the unrequired features in order to lessen waste [40].

The Kanban method employs visual Kanban board to improve software development. Here, the development process is displayed in its different stages. As noted, Kanban is less prescriptive than Scrum because it has no defined roles, no stress on meetings, and no artifacts [4].

TABLE I
SCRUM PRACTICES DEFINITIONS.

Practices, Artifacts, and Activities of Scrum	Definition
Product Backlog [21]	A list containing the complete features and requirements for implementation [21],[30],[31]. The product backlog is constructed and maintained by the product owner. The priority and projected effort are appropriated using the items/features of the product backlog [21],[32].
Sprint Backlog [21]	A collection of activities to be executed during a sprint [21],[30],[31]. Each activity or set of activities comprises of product backlog item's representation which is to be added into the following release which will be presented, the moment the current sprint ends [21],[30].
Sprint Burndown Chart [21]	A chart is representing the remainder of sprint backlog work to be updated every day, within which progress is measured against the goals of iteration [21],[33]. Usually, experts need to estimate the completion of the tasks/stories that remain for each sprint backlog [21].
Effort Estimation [21]	All members that make up the team that employs Scrum methods and the members utilize the difficulty level and effort level for estimation of work [21],[30]. Effort scaling which entails an abstracted metric is also employed, such as t-shirt sizing or numeric sizing [21].
Sprint [21]	Sprint is the core of Scrum and comprises a time-box with the duration of up to one month. A finished, functioning and releasable product are formed inside the duration [21],[22]. The outcomes show that most informants that employed Scrum remained within a duration sprint of two weeks [21]. Nonetheless, a sprint of four or six months is possible as well [22], [30]. Also, after the sprint to be followed has been determined by the team, no changes can be made, but in certain situations, the product owner can revoke the chosen sprint even though resources may be incurred [21].
Sprint Planning [21]	Sprint planning is mainly for deciding on the conveyable items within the present sprint of the duration between 2-4 weeks duration, and on how the given task is to be accomplished for the delivery of the sprint [21],[22].
Daily Stand-up Meetings [21]	Comprise the daily 15-minute meetings on project progress analysis, with feedback, both positive and negative [21]. During the meeting, unanticipated matters which could cause project delay are addressed together with the work to accomplish before the following meeting [21],[22],[34]. For teams that are large, daily Scrum will not offer the members of the team the value anticipated [22],[35].
Sprint Review [21]	Team members and the customer meet for review following each sprint. The meeting involves all stakeholders. During the meeting, the new release features are all highlighted [21],[30]. Customer feedback during the sprint review meeting appears to increase the quality of each sprint [36], making both customer preferences and feedback essential [37],[38].
Sprint Retrospective [21]	This practice promotes discussion among team members on the issues faced and how these issued can be tackled in future sprints [21],[34]. Sprint Retrospective allows better effectiveness to developers by way of behavioral modification by the encountered situations [21],[34],[36].

In the context of Kanban, visualization of workflow is via the Kanban board. This restricts work in progress via the minimization of the number of features to be implemented, flow measurement and management, the creation of clear

policies, feedback implementation, and consistent looping and collaboration improvement [23]. Accordingly, Table II details the practices of Kanban.

TABLE II
KANBAN PRACTICES DEFINITIONS.

Practices	Definition
Visualize the workflow [23]	This practice involves making the necessitated work visible to the entire team members considering that invisible work can lead to project implementation risk. The Kanban board provides clear visualization of the workflow [23],[27].
Limit work in progress [23]	The key practice in Kanban; it relates to the restriction to work in progress (WIP) regarding the amount or restricting the features of WIP [23],[41]. The reduction of the amount of multitasking causes the reduction of the time of features delivery [6],[23].
Manage and measure flow [23]	The main purpose of this practice is to complete the task at hand. Here, all states within the workflow are supervised. This method is dubbed as measuring flow. Such supervision generates efficient and smooth movement, which generates appropriate value. This reduces risk and prevents postponement cost [6],[23].
Make policies explicit [23]	Given the countless forms of work to be dealt with, most firms have implicit policies. On the other hand, for their constant usage, the Kanban teams explicitly present the policies [6],[23],[27].
Implement feedback [23]	Kanban requires feedback loops in order to function [23],[41]. Hence, it employs a standup meeting, in addition to a review of service delivery, operations, and risk. This allows the comparison between the projected and the real outcomes, and the execution of adjustments needed [23].
Continuous improvement [23]	This practice entails clarified and universal awareness of theories that relate to work, workflow, process as well as risk. Such awareness assists the members in the development of a mutual discernment regarding a problem while also proposing a solidly agreed improvement [23]. A Kaizen mentality should be embraced by the team [42]-[44], as it is a key part of the practical application of Kanban.

C. Scrumban Method

Scrumban is a hybrid Agile method of Scrum and Kanban[45]-[47]. Agile team members look for the appropriate thing to work on next, no more and no less. Hence, a hybridization of Scrum and Kanban is needed to enhance Scrum method by omitting its inappropriate practices and by adopting appropriate practices from the Kanban method [45]. Hence, team members adopt the appropriate practices of both methods based on different situations to meet the needs[48],[49]. Scrumban supports Agile team members to be creative in developing new methods to meet their needs [47]. Moreover, there are no specific practices for Scrumban, but the Agile team members have to understand, which practices of Scrum and Kanban deliver value and choose the appropriate practices accordingly [48]. Scrumban is more adaptive especially when there are changes in user requirements [50]. It was found that Agile team members, who are familiar with Scrum and Kanban can combine and benefit more from Scrumban [49],[50]. The formation of Scrumban practices and definition of them are presented in Table III as follows.

TABLE III
SCRUMBAN PRACTICES FORMATION.

Practices, Artifacts, and Activities of Scrumban	Applicability and Definition
Product Backlog	Just in time stories, there is no need to emphasize on having all the stories [45],[50].
Sprints (Sprint Planning, Sprint Retrospective and Sprint Review)	Scrumban focuses more on continuous flows instead of Sprints [45],[51]. However, Scrumban team members can decide which practices are appropriate for them and adopt them accordingly [51]. Unlike Scrum, the changes can be done during the sprint development [51],[52].
Effort Estimation	Unlike Scrum, effort estimation is not compulsory, and the team would decide whether to estimate or not [45],[51],[50].
Roles	Scrumban does not emphasize having all the roles, and the team would decide which roles are appropriate to be adopted [50]-[52]. For instance, when the team size is very small, they can decide whether to have a Product Owner or Scrum Master. It is also not compulsory to have a cross-functional team, and team members can be specialized in different areas [50].
Daily Meeting	When needed only, just to make sure there is a continuous improvement [45],[47],[50],[51].
Sprint Burn down Chart	Not required and Scrumban team member can use extended board by dividing the work in progress column to show the details progress of every phase [45],[50].
All Kanban Practices as needed	Visualize the workflow, continuous improvement, manage and measure flow, limit work in progress, make policies explicit and implement feedback are the main practices of Kanban and all of them can be adopted as needed when forming Scrumban [50]-[52].

D. Materials and Method

The main highlights of this research are the Kanban, Scrum and Scrumban methods in addition to the determination of their feasibility in projects. Hence, to identify the pertinent and accessible scholarly works in elucidating the methods' criteria, a review was carried out. Furthermore, the identification and scrutiny of the selection factors and their usage in the selection and combination of both methods were made possible through interviews. This is followed by the discussion of the differences and resemblances of both methods. Additionally, factors impacting the choosing of both methods were discussed. The review was carried out in accordance to keywords, type of study, the language used, and publication year.

In the search, the keywords comprise the factors and criteria of selection, Scrum, and Kanban. Regarding the criteria of inclusion for each category, the used key points are as follows:

- Studies that detail the Scrum, Kanban and Scrumban methods.
- Studies that demonstrate how Scrum and Kanban can be combined.
- Studies that delve into the practices about the Kanban, Scrum and Scrumban methods.
- Finally, studies that delve into the factors that affect the selection and combination of the practices of the three methods.

The earliest studies chosen in this study are those from year 2001; this is the year when Agile methods started

gaining popularity among the research industry and community. In addition, the present study adopted semi-structured in-depth interviews with open-ended questions to help the researcher to obtain and understand the views, experiences, perceptions and viewpoints of the Agile experts on Scrum and Kanban selection and combination. Seven experts were chosen from different organizations from different countries, among which 2 were Malaysian (VP of engineering and chief executive officer), 1 was Singaporean (Director of Agile organization), 1 was Palestinian (Senior Agile developer), 1 was from the United States (Principal consultant), 1 from Canada (Senior consultant), and lastly, 1 was from New Zealand (Chief knowledge engineer). The participants were given a number from P1-P7 for anonymity purposes. Each interview was recorded using an audio recorder after taking the approval of the respondents, with every interview session averaging 66 minutes. A set of questions were designed to understand the selection of the methods. In addition, since the interviews were semi-structured with probing questions, the researchers managed to investigate how Scrumban can be formed. The selected studies and the interview transcripts were analysed qualitatively by the use of content analysis. NVivo software was used to organize the gather data. Content analysis employs a summary as well as analysis of the gathered data to make a comprehensive comparison between the Scrum, Kanban and Scrumban methods. It also comprises the direct words compression within a text. Hence, following the rules of coding, the categories created were less number-wise [53]. During coding, each segment of text is labelled, and the text range could be from a few words to the whole paragraph. Coding enables the re-arrangement and integration of words, sentences or paragraphs that are interrelated. Hence, meaningful data could be created [53]. Accordingly, the outcomes of this study include the comparisons of Scrum, Kanban and Scrumban, in addition to the factors influencing the selection of the practices of each method. In addition, it demonstrates how Scrumban method can be formed.

III. RESULTS AND DISCUSSION

The methods of Kanban and Scrum are both usable in different circumstances (refer to Table IV). Also, the review demonstrates the expansive usage of Scrum method, more than that of Kanban [8]-[10]. Relevantly, Table IV highlights the dissimilarities between both methods as discussed in the extant literature. Also, Scrumban method is highlighted in the same table.

As opposed to Kanban, Scrum entails a prescribed method with details that are significant for an Agile team [3],[4],[21]. Meanwhile, in Kanban, team size appears to have greater flexibility; the batch size required by Scrum based on sprint while Kanban needs fairly small batch size, even as minute as hourly batch [5]. Further, Kanban enables daily prioritized requirements, whereas when following Scrum, the requirements prioritization must be grounded upon the length of the given sprint [5]. Additionally, Kanban has very small feature size, as opposed to Scrum [6]. Moreover, the use of Kanban rather than Scrum causes the lead time or the time length between the proposal of a fresh feature or a request creation, in addition to its delivery to the

customer's setting for the related organization, to go down by half (50%) [6]. Also, inaccurate estimates and timeboxes have been reported to cause longer lead times, leading to waste [28]. Furthermore, the use of Kanban reduces cost,

which means that Kanban is more cost effective as opposed to Scrum [10] particularly in the context of operations [15]. The use of Kanban also improves quality [15][28]. However, both methods have no technical practices.

TABLE IV
SCRUM, KANBAN, AND SCRUMBAN DIFFERENCES.

Criteria	Scrum	Kanban	Scrumban
Prescription	Prescribed [21].	Not prescribed and flexible [3],[5].	Not prescribed and flexible [45],[47]
Roles & Responsibilities	Predefined [7],[21],[41],[44],[54].	Not predefined [4],[5],[44].	Based on the team decision [48],[50],[51].
Adoption time	The transition is a bit challenged [6]. However, Agile organizations seem to adopt Scrum before Kanban [15].	Transition to Kanban is easier [6], especially when teams migrate from structured methods such as waterfall [15].	Team experience in Scrum and Kanban is needed to be able to combine them [45],[51].
Team size	A team is containing 5-11 members with predefined roles [21],[22]. If the team is large, scaling is needed [25]. Hence, for large teams, Kanban could better than Scrum [33].	More flexible than Scrum when considering the team size [33]. Team members could be less than 5 members [55], or it could be more than 11 (up to 14 team member) [33].	More flexible when compared to Scrum and somehow similar to Kanban [45],[51].
Batch Size (WIP size)	The batch size is large in comparison to Kanban [5],[15], and teams are required to deliver sprints on time [5],[41],[54].	The batch size of the requirements is small [5],[6],[15],[54]. Daily/hourly delivery of urgent items can be done [5], and commitment is not compulsory [5],[41],[54].	Based on the team decision [45],[47].
Requirements prioritization	Requirements prioritization is based on the length of the sprint [5],[44].	Requirements prioritization is done continuously, which can be daily or hourly [5],[44].	Based on the team decision [45],[51].
Feature size	Small-feature size [6],[15].	The feature size is smaller when compared to Scrum [6],[15].	Based on team decision [51].
lead time	Scrum avoids cutting lead time unlike Kanban [28].	Kanban cuts lead time by steering clear of multi-tasking and limiting the WIP features [15],[16],[29],[41].	Short and considered to be better than Scrum and Kanban [46],[47],[51].
Technical practices	No technical practices [44],[56].	Kanban also has no technical practices [39],[57]	Like Scrum and Kanban, Scrumban has no technical practices [45],[51].
Cost	Unlike Kanban, Scrum avoids cost saving, but it focuses more on knowledge, experience and decision making based on what is known [10]	Focuses on cutting cost [10], especially for operations [15],[58].	Focuses more on cutting cost when compared to Scrum and Kanban [49].
Quality	Sprint review meeting is the main practice for improving quality in the Scrum method [59]	Kanban focuses more on improving quality when compared to Scrum [15],[28].	Quality will be increased when using Scrumban [51].

Based on Table IV above and the empirical evidence, the following cases illustrate the appropriateness of each method and how the selection of Kanban, Scrum and Scrumban practices could be done.

Case 1: The Selection of Kanban

When the practitioners of Agile demonstrate no preference in adopting the recommended Agile methods, it implies their non-preference towards adhering to the roles that were predefined. These practitioners could simply opt for Kanban instead. Kanban would be optimal for teams with 3-14 development members who could not optimally batch the work into sprints of 1, 2, or 4 weeks. Kanban is also suitable for team members who are required to perform batching or prioritization every day or every hour. In Kanban selection, the features are classable into portions of smaller size, whereas value can be delivered every day or every hour, which means that it is deliverable in less than one week time. Kanban is more fitting for practitioners who need to concentrate more on lead time reduction, quality enhancement, and cost reduction. The above criteria for selecting Kanban were supported by different Agile experts as follows:

- **Roles & Responsibilities:** A chief knowledge engineer has mentioned: "my team members are

specialists; they do not want to change their identity to have new roles and responsibilities. So, they prefer to adopt Kanban" R1.

- **Team size:** As mentioned by a senior consultant: "Kanban is more flexible than Scrum, you can have 3 Agilists working together as a team, or you can have more than 10" R5.
- **Batch Size:** As mentioned by a director of the Agile organization: "If the cadence of work is less than two weeks, the request coming must be resolved. Then we will use Kanban. If however it can be batched, then we will use SCRUM" R2
- **Requirements prioritization:** A chief knowledge engineer has mentioned: "We have to limit WIP since the requirements are changing dramatically on an hourly basis." R1
- **Feature size:** As mentioned by a senior agile developer: "If the features are small and not complex, then just manage the flow and limit work in progress, this is how Kanban works." R3
- **Lead time:** As mentioned by VP of Engineering: "If you want fast delivery, Kanban is the right choice." R4

- *Cost*: As mentioned by chief information officer: “Kanban is cost-effective in comparison to Scrum.” R6
- *Quality*: As mentioned by principal consultant: “Kanban will assist more in running and solving quality issues” R7

Case 2: The selection of Scrum

When practitioners of Agile demonstrate preference in the adoption of the methods recommended, they will adhere to the roles and responsibilities prescribed. Hence, the shift to Scrum would appear simple for them. Scrum is usually appropriate for teams comprising of 5-11 development members, while the size of the batch is based on a weekly sprint; this is deemed as simple, estimation-wise. Scrum would be the appropriate option for requirements of prioritization flows into a weekly sprint, whereas the size of the feature is characterized by size with small feature doable in one, two or four weeks sprint. Notably, Scrum in comparison to Kanban has no regard over lead time and cost reduction. Rather, the focal points of the team are more towards knowledge, experience and decision making grounded upon what is recognized. Scrum also does not take into account the issue of quality, in-depth. The above criteria for selecting Scrum were supported by different Agile experts as follows:

- *Roles & Responsibilities*: A director of the Agile organization has mentioned: “If roles and responsibilities make things clear for all, then Scrum is appropriate” R2.
- *Team size*: As mentioned by VP of Engineering: “Following the roles and responsibilities of Scrum, the best team members should be 7 ± 2 . So, the maximum is 9 or 10, and the minimum should be 5 members.” R6.
- *Batch Size*: As mentioned by a director of the Agile Organization: “If the cadence of work is less than two weeks, the request coming must be resolved. Then we will use Kanban. If however it can be batched, then we will use SCRUM” R2
- *Requirements prioritization*: A chief knowledge engineer has mentioned: “Requirements prioritization is done based on the length of the sprints, so for each sprint backlog, we have to select the requirements which provide values and batch them into weeks sprint.” R1
- *Feature size*: As mentioned by a senior agile developer: “For Scrum, the features are still small but there is a need to estimate their size to be able to decide what you are going to deliver within two weeks’ time.” R3
- *Lead time*: As mentioned by VP of Engineering: “In my opinion, Kanban assists the team to deliver value and get regular feedback faster than Scrum.” R4
- *Cost*: As mentioned by chief information officer: “Kanban is cost-effective in comparison to Scrum.” R6
- *Quality*: As mentioned by principal consultant: “Kanban will help more in running and solving quality issues” R7

Case 3: Scrumban Formation

Based on case 1 and 2, it is difficult to be limited to a single method since it is so difficult to satisfy all the criteria for the development of a specific project. Therefore, a hybridization of Scrum and Kanban is needed. To hybridize both methods successfully, there is no single case for selecting Scrumban, and the formation of it will be based on the situation at hand. For instance, Agile team members could initiate the project by adopting Scrum. However, the team could avoid sprints, sprint planning, sprint review, and sprint backlog when the work batch size is too large/too small to fit in a sprint and the estimation of the sprints size or duration is difficult. In this case, the team would keep the other practices of Scrum and adapt the method by adopting other practices from Kanban. Example of Kanban practices which are suitable for this situation is limiting WIP and visualizing the workflow. Also, another team could initiate the project by using Scrum but when the prioritization of work or requirements are done daily or hourly, then managing and measuring the flow and limiting work in progress practices are needed.

On the other hand, when the team is too small, less than 5 members, then the team members might not stick to having all the predefined roles as defined by Scrum. Based on the above examples, it is evident that Agile team members are responsible for forming the Scrumban method based on different situations. However, they should understand the practices of both methods to be able to select the appropriate practices. The above criteria for combining both methods were supported by different Agile experts as follows:

- *Roles & Responsibilities*: A director of the Agile organization has mentioned: “The development team and the management would decide which roles are needed” R2.
- *Team size*: As mentioned by VP of Engineering: “Scrumban team size is just like Kanban but Kanban can be more flexible.” R6
- *Lead time*: As mentioned by VP of Engineering: “In my opinion, Kanban assists the team to deliver value and get regular feedback faster than Scrum.” R4
- *Cost*: As mentioned by chief information officer: “The primary focus of Scrumban is to decrease the cost and waste by adopting the best practices from both methods.” R6
- *Quality*: As mentioned by principal consultant: “During the daily stand-ups, Scrumban focuses on how to make continuous improvement. So, it takes advantages of both methods to increase the quality” R7

Also, Scrumban allows the team members who got experience in adopting both methods to decide how to limit WIP. As mentioned by a senior agile developer: “If you have experience in using both methods, you can decide the length of your sprints, the limit the work in progress accordingly” R7.

IV. CONCLUSIONS

The use of different methods such as Kanban and Scrum in software development presents challenges to the organizations that adopt Agile methods. It is challenging to select one method over the other (Scrum over Kanban, or vice versa) or both methods in hybrid form (Scrumban) during the implementation of a given project. Hence, during the selection of the Scrum and Kanban methods or the hybridization of both, a range of criteria has to be considered. Agile team members should be assisted when making this crucial decision especially when they are planning to hybridize both methods. Thus, an in-depth review of the literature was conducted in this study to explore the selection factors for Scrum and Kanban and how the selection factors could assist in taking the right decision when forming Scrumban method. Also, interviews were conducted to investigate the phenomena in more details. Both Kanban and Scrum complement each other. However, the method prescription, roles, and responsibilities, adoption time, team size, batch size, requirements prioritization, feature size, lead time, technical practices, cost, and quality are the major factors that assist the Agile team members in the formation of Scrumban by selection of the appropriate practices from both methods. Minimized waste and the likelihood of project delay [60], improving the quality of the delivered products and applying the Kaizen mind to continuously make improvements are the main benefits of Scrumban [45],[51],[50]. Different case studies on the selection of Scrum, Kanban and Scrumban were provided to show how Scrumban can be successfully formed based on Scrum and Kanban practices.

ACKNOWLEDGMENT

We are indebted to the Faculty of Information Science and Technology at Universiti Kebangsaan Malaysia (UKM) for providing support to this research under the Fundamental Research Grant Scheme project (FRGS/2/2013/ICT01/UKM/02/2).

REFERENCES

- [1] A. Alliance, "Agile Manifesto," *Online at <http://agilemanifesto.org/>*, vol. 6, no. 1, 2001.
- [2] S. W. Ambler, "The Non-Existent Software Crisis: Debunking the Chaos Report," 2014. [Online]. Available: <http://www.drdoobs.com/architecture-and-design/the-non-existent-software-crisis-debunki/240165910>.
- [3] N. Nikitina and M. Kajko-Mattsson, "Developer-driven big-bang process transition from Scrum to Kanban," in *Proceedings of the 2011 international conference on software and systems process*, 2011, pp. 159–168.
- [4] H. Kniberg and M. Skarin, *Kanban and Scrum-making the most of both*. Lulu. com, 2010.
- [5] R. Cuellar, "Kanban for Help Desks: Managing the Unplannable," *Cut. IT J.*, vol. 24, no. 3, p. 23, 2011.
- [6] A. Shalloway, "Demystifying Kanban," *Cut. IT is J.*, vol. 24, no. 3, p. 12, 2011.
- [7] R. H. Al-Ta'ani and R. Razali, "A Framework for Requirements Prioritisation Process in an Agile Software Development Environment: Empirical Study," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 6, no. 6, pp. 846–856, 2016.
- [8] A. M. M. Hamed and H. Abushama, "Popular agile approaches in software development: Review and analysis," in *Computing, Electrical and Electronics Engineering (ICCEEE), 2013 International Conference on*, 2013, pp. 160–166.

- [9] R. V. Anand and M. Dinakaran, "Popular Agile Methods in Software Development: Review and Analysis," *Int. J. Appl. Eng. Res.*, vol. 11, no. 5, pp. 3433–3437, 2016.
- [10] A. S. Campanelli and F. S. Parreiras, "Agile methods tailoring—A systematic literature review," *J. Syst. Softw.*, vol. 110, pp. 85–100, 2015.
- [11] R. H. AL-TA'ANI and R. Razali, "Process Model For Systematic Requirements Prioritisation Process In An Agile Software Development Environment Based On 5S Approach: Empirical Study," *J. Theor. Appl. Inf. Technol.*, vol. 95, no. 8, 2017.
- [12] R. Hoda and J. Noble, "Becoming Agile: A Grounded Theory of Agile Transitions in Practice," in *Proceedings - 2017 IEEE/ACM 39th International Conference on Software Engineering, ICSE 2017*, 2017.
- [13] O. N. A. Al-Allaf, "The Adoption of Agile Processes in Large Web Development Enterprises: A Survey in Jordan," *IJWA*, vol. 2, no. 3, pp. 206–216, 2010.
- [14] F. Ahmad, F. Baharom, and M. Husni, "Agile development methods for developing a web application in small software firms," in *Knowledge Management International Conference (KMICe)*, 2012.
- [15] D. Verweij and O. Maassen, "Kanban at an Insurance Company in the Netherlands," *Viral Growth Kanban Enterp. Cut. IT J.*, 2011.
- [16] D. J. Anderson and A. Rock, "An agile evolution: why Kanban is catching on in Germany and around the world," *Cut. IT is J.*, vol. 24, no. 3, p. 6, 2011.
- [17] S. Denning, "Why Agile can be a game changer for managing continuous innovation in many industries," *Strategy. Leadersh.*, vol. 41, no. 2, pp. 5–11, 2013.
- [18] M. Alqudah and A. Abdulsalam, "Implementing computer-aided language learning tool using the hybrid agile method: A case study," in *Proceedings - 2013 International Conference on Informatics and Creative Multimedia, ICICM 2013*, 2013.
- [19] M. Alqudah and A. Abdulsalam, "Basic English language tools for beginners: Using animations and audio," *Int. J. Sci. Eng. Res.*, vol. 4, no. 4, pp. 228–233, 2013.
- [20] L. Abdulwahab, A. A. Abdalla, B. S. Galadanci, M. Algudah, and M. Murtala, "Agile Methods for Software Engineering Students Project: A Proposed Hybrid Methodology," in *The Third International Conference on Digital Enterprise and Information Systems (DEIS2015)*, 2015, vol. 63.
- [21] J. Sutherland and K. Schwaber, "The scrum guide. The definitive guide to scrum: The rules of the game," *Scrum.org October* 2013.
- [22] J. López-Martínez, R. Juárez-Ramírez, C. Huertas, S. Jiménez, and C. Guerra-García, "Problems in the Adoption of Agile-Scrum Methodologies: A Systematic Literature Review," in *Software Engineering Research and Innovation (CONISOFT), 2016 4th International Conference in*, 2016, pp. 141–148.
- [23] D. J. Anderson, "Kanban." Blue Hole Press Sequim, WA, 2010.
- [24] Z. Mansor, S. Yahya, and N. H. Arshad, "Towards the Development of Success Determinants Charter for Agile Development Methodology," *Int. J. Inf. Technol. Eng.*, vol. 2, no. 1, pp. 1–7, 2011.
- [25] M. Alqudah and R. Razali, "A review of scaling agile methods in large software development," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 6, no. 6, 2016.
- [26] A. A. Hamid and Z. Mansor, "Client's Readiness Assessment Success Factors for Outsourcing Software Projects," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 6, no. 6, 2016.
- [27] H. Lei, F. Ganjeizadeh, P. K. Jayachandran, and P. Ozcan, "A statistical analysis of the effects of Scrum and Kanban on software development projects," *Robot. Comput. Integr. Manuf.*, vol. 43, pp. 59–67, 2017.
- [28] D. I. K. Sjøberg, A. Johnsen, and J. Solberg, "Quantifying the effect of using kanban versus scrum: A case study," *IEEE Softw.*, vol. 29, no. 5, pp. 47–53, 2012.
- [29] S. Govindaraj and S. Tadipatri, "Use of Kanban in distributed offshore environments," *Viral Growth Kanban Enterp. Cut. IT J.*, 2011.
- [30] V. T. Heikkilä, M. Paasivaara, and C. Lassenius, "Scrumbut, but does it matter? a mixed-method study of the planning process of a multi-team scrum organization," in *Empirical Software Engineering and Measurement, 2013 ACM/IEEE International Symposium on*, 2013, pp. 85–94.
- [31] S. Ashraf and S. Aftab, "Latest Transformations in Scrum: A State of the Art Review," *Int. J. Mod. Educ. Comput. Sci.*, vol. 9, no. 7, pp. 12–22, 2017.
- [32] C. Kaur1st and V. Kumar 2nd, "Product Backlog Prioritization in Scrum: A Review," *Int. J. Mod. Comput. Sci.*, vol. 3, no. 2, 2015.

- [33] V. Moskalenko, "Scrum vs. Kanban: Which to Choose for Agile Development?" 2015. [Online]. Available: <https://www.luxoft.com/blog/vmoskalenko/scrum-vs-kanban-which-to-choose-for-agile-development/>.
- [34] Y. Andriyani, "Knowledge Management and Reflective Practice in Daily Stand-Up and Retrospective Meetings," in *International Conference on Agile Software Development*, 2017, pp. 285–291.
- [35] V. G. Stray, Y. Lindsjorn, and D. I. K. Sjoberg, "Obstacles to efficient daily meetings in agile development projects: A case study," in *Empirical Software Engineering and Measurement, 2013 ACM/IEEE International Symposium on*, 2013, pp. 95–102.
- [36] A. Khosravi, T. J. Gandomani, and H. Fahimian, "Introduction of a scrum in an elite team: A case study," *J. Softw.*, vol. 12, no. 3, pp. 173–180, 2017.
- [37] M. Alqudah, Y. Yusof, S. A. M. Noah, and A. Almbahouh, "Incorporating Prioritized User Preferences in Search System," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 2, no. 5, pp. 401–404, 2012.
- [38] M. K. Al-Qudah, "A Weighted-Based Approach To Prioritize User Preferences In Information Retrieval," Universiti Utara Malaysia, 2008.
- [39] A. Bolaji, "A cross-disciplinary systematic literature review on Kanban," Master's Thesis. The University of Oulu. 62 p. Available at: <http://jultika.oulu.fi/files/nbnfioulu-201502111073.pdf>, 2015.
- [40] T. Skeie, "Does Limit on Work-In-Progress (WIP) in Software Development Matter?" 2014.
- [41] M. O. Ahmad, J. Markkula, and M. Oivo, "Kanban in software development: A systematic literature review," in *Software Engineering and Advanced Applications (SEAA), 2013 39th EUROMICRO Conference on*, 2013, pp. 9–16.
- [42] B. Estácio, R. Prikładnicki, M. Morá, G. Notari, P. Caroli, and A. Olchik, "Software kaizen: Using agile to form high-performance software development teams," in *Agile Conference (AGILE), 2014*, 2014, pp. 1–10.
- [43] O. Badreddin, "Empirical evaluation of research prototypes at variable stages of maturity," *2013 2nd Int. Work. User Eval. Softw. Eng. Res. USER 2013 - Proc.*, pp. 1–4, 2013.
- [44] G. S. Matharu, A. Mishra, H. Singh, and P. Upadhyay, "Empirical study of agile software development methodologies: A comparative analysis," *ACM SIGSOFT Softw. Eng. Notes*, vol. 40, no. 1, pp. 1–6, 2015.
- [45] C. Ladas, "Scrumban," *Lean Softw. Eng. Contin. Deliv. High Qual. Inf. Syst.*, 2008.
- [46] M. Yilmaz and R. O'Connor, "A scrumban integrated gamification approach to guide software process improvement: a Turkish case study," *The. Vjesn. (Technical Gazette)*, vol. 23, no. 1, pp. 237–245, 2016.
- [47] A. Reddy, *The Scrumban [r] evolution: Getting the Most Out of Agile, Scrum, and Lean Kanban*. Addison-Wesley Professional, 2015.
- [48] H. Dunsky, *A case for Scrumban*. 2014, pp. 1–10.
- [49] J. Similä, M. Oivo, and K. Liukkunen, "Empirical Investigation of Scrumban in Global Software Development," in *Model-Driven Engineering and Software Development: 4th International Conference, MODELSWARD 2016, Rome, Italy, February 19-21, 2016, Revised Selected Papers*, 2017, vol. 692, p. 229.
- [50] Aniruddha Joshi and M. Sandeep, "Our Journey into Scrumban," 2014. [Online]. Available: <https://www.arrkgroup.com/thought-leadership/our-journey-into-scrumban/>. [Accessed: 20-Jul-2017].
- [51] S. Pahuja, "What is Scrumban," *Agile Alliance*, 2015. [Online]. Available: <https://www.agilealliance.org/what-is-scrumban/>. [Accessed: 20-Jul-2017].
- [52] A. Banijamali, R. Dawadi, M. O. Ahmad, J. Similä, M. Oivo, and K. Liukkunen, "An empirical study on the impact of Scrumban on geographically distributed software development," in *Model-Driven Engineering and Software Development (MODELSWARD), 2016 4th International Conference on*, 2016, pp. 567–577.
- [53] K. Krippendorff, *Content analysis: An introduction to its methodology*. Sage, 2012.
- [54] M. O. Ahmad, P. Kuvaja, M. Oivo, and J. Markkula, "Transition of software maintenance teams from Scrum to Kanban," in *System Sciences (HICSS), 2016 49th Hawaii International Conference on*, 2016, pp. 5427–5436.
- [55] P. Brodzinski, "When Kanban is the Best Choice," 2010. [Online]. Available: <http://brodzinski.com/2010/06/kanban-best-choice.html>.
- [56] S. W. Ambler and M. Lines, "Going Beyond Scrum: Disciplined Agile Delivery," *Discip. Agil. Consortium. White Pap. Ser.*, 2013.
- [57] A. Bolboaca, "Agile Challenge: Adopting Technical Practices," *Mozaic works*, 2013. [Online]. Available: <https://mozaicworks.com/blog/agile-challenge-adopting-technical-practices/>. [Accessed: 01-Mar-2016].
- [58] M. O. Alqudah and R. Razali, "Key factors for selecting an Agile method: A systematic literature review," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 7, no. 2, 2017.
- [59] F. Kanwal, K. Junaid, and M. A. Fahiem, "A hybrid software architecture evaluation method for fdd-an agile process model," in *Computational Intelligence and Software Engineering (CiSE), 2010 International Conference on*, 2010, pp. 1–5.
- [60] M. Alqudah and R. Razali, "A comparison of scrum and Kanban for identifying their selection factors," in *Electrical Engineering and Informatics (ICEEI), 2017 6th International Conference on*, 2017, pp. 1–6.