Scientific Journal of Impact Factor (SJIF): 5.71

E-ISSN (O): 2348-4470
P-ISSN (P): 2348-6406

International Journal of Advance Engineering and Research Development

Volume 8,Issue 11,November 2021

SCRUMBAN : The Optimal Approach Using Scrum and Kanban

Shweta Manohar Waghmare

Department MCA,VJTI College

Abstract —*Scrumban may be a combination of practices from Kanban and Scrum practices to manage the event of software supported different project situations. Scrumban practices formation can differ supported different projects and team members. However, since each method has its own pros and cons, inappropriate formation of Scrumban practices may cause increased waste and time of development, and decreased quality, which successively, affect the Agile organizations and cause inefficient and ineffective development. This study aims to show how Scrumban method is created supported a combination of Kanban and Scrum using task priority approach and indivisual team effort. Scrumban formation and therefore the identification of the factors, which assist within 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. various factors - the strategy prescription, roles and responsibilities, adoption time, team size, batch size, requirements prioritization, feature size, lead time, technical practices, cost and quality, assist Additionally, Scrumban were found to be more appropriate than Scrum or Kanban in measuring task priority, saving time, improving quality and minimizing waste*

Keywords*- Scrumban , Agile, Kanban, Scrum, software development .*

1. INTRODUCTION

Agile method is that the underpinning conception to a collection of software development methods. It corresponds to four values and twelve principles of the Agile Manifesto. Scrumban method, which may be a Hybrid of Scrum and Kanban methods, was invented by these values and principles. The values above specialise in working software, individuals and interactions, customer collaboration, and reacting to vary quickly. The captured values are refined into 12 principles as shown within the alliance's manifesto .Several of the principles are as follows:

- Stakeholders (users and customers) and developers must cooperatively work during the project.
- Projects must be constructed around individuals who are driven.
- The direct discussion is that the simplest and efficient method for information transmission within the team of development and delivery.
- Working software is that the most appropriate method for measuring progress because it quantifies the worth of the business.
- A relentless pace should be maintained to enable sustainable delivery and development

Key Benefits of agile software development methodology are as follows-

- Working software
- Customer Collaboration
- Quick response to change
- Individual and Interaction

In case of dynamically changing requirement, delivering software as fast as possible with prime quality product within shorter period besides customer satisfaction, so that industry needs better method to achieve service. By implementing agile software methodologies all software development demands are achievable

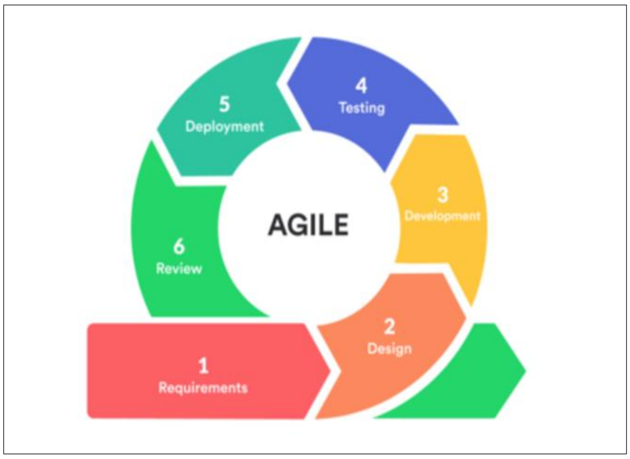

Figure 1. Agile Process

The main of the study is to overcome the challenge in scrumban which is come from scrum and kanban. Scrumban derives some set of useful practices from Scrum and Kanban; and constitutes a strong framework that revolves agile transformations to deal with unpredictable problems also. Scrum has capabilities to handle complex challenges; while Kanban is incredibly simple and not rigid and may be fit with the Scrum rule and regulations. Scrum is incredibly suitable for wares development while Kanban is suitable for information and work flow management for normal production industries but still flow based approach of Kanban is more convenient than time boxed
approach of Scrum. The mix of Scrum and Lean based Kanban has been found very effective in Software Engineering Management. Even if kanban is best for visualization still it lack some rigidity and where scrum is used everywhere.

II. AIM AND OBJECTIVES

A) AIM

The main aim of this approach is to overcome challenges in scrumban like individual team effort , assigning task according to the performance key factor

B) OBJECTIVE

The main objective of this approach is to present better look and adding efficiency to software and finding priority of task based on the how task or project is urgance (high, medium, low level) As task priority is find at planning event, there is need to take some factor before assign priority. For each method as scrum, kanban and scrumban it has different way of visualization. Here it simplifies the visualization

C) SCRUM

Scrum is widely used methodology. As it is used everywhere. It uses iterative and incremental approach which is simple to understand and with little duration products.
It is best suited for complex and huge system.
There is concept of sprint which is of 1 week or 2 week or 4 week. For example : facebook or any other social platform which is widely used . Facebook is already build but for making change for scrum can be used. Scrum is used as small spring as small section of each facebook like feeds, chats, comments, post, user details etc.
In scrum there is concept of product backlog which congaing multiple requirement prioritized by the business value. After that there is sprint planning to know how much it can commit to deliver by the end of sprint.
Sprint Backlog is that the collection of Product Backlog items selected for the Sprint, plus a thought for delivering the Increment and realizing the Sprint Goal.

Daily meeting is conducted to know the progress of each sprint or task .
Team member respond to basic :

- What did you do still last meeting ?
- Do you have any obstacles?
- What will you do before nextt meeting?

Sprint Retrospective is for :

- What well worked ?
- What could be improved ?
- What will we commit to doing in next sprint?

Sprint Review is meeting that involve product owner, scrum master, team and others members that discussing about :

- What functionalities “Done” ?
- What is next task ?

- Next sprint planning date ?

Backlog refinement is ongoing process for adding details, estimate and in order to item in product backlog.It check following ;

- Is the Why clear?
- Is the What clear?
- Contribute to product vision?
- Does it add value ?

After all successful meeting and good planning and estimation product is release in the term of “Potentially shippable product increment”

Roles involved in Scrum are Product Owner, Scrum Master, and Scrum Team. Product owner is responsible to get the input as requirements from stakeholders, Customer, End User, etc. Product owner create product backlogs using these requirements. Product owner only can make necessary changes in product backlog. Scrum Team ideally forms with 7±2 members i.e. group of 5 to 9 members. In case of large products, multiple scrum tams get involved. Members in scrum team are typically ‘cross functional’ like designers, coders, testers. There is no role of manager in scrum project. Scrum team use to give status in scrum meeting to Scrum Master.

- 90% - Estimated Agile team use Scrum
- 12M+ Estimated Using Scrum daily
- Practiced everywhere

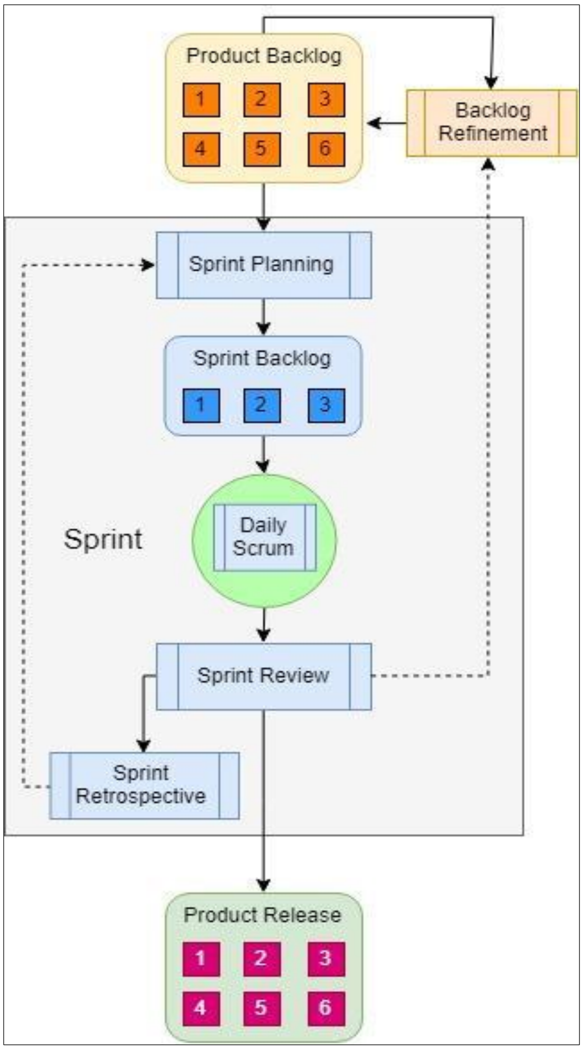

Figure 2. Basic Scrum flow

IV. KANBAN

Kanban is meant for transparent workflow management by visualizing current state of each work item for the objective of scheduling system for lean and just-in-time manufacturing at Toyota by Taiichi Ohno with the goal of limit WIP (Work-in-Progress).

Principles of Kanban –

- Visualize flow of work
- Limit the amount of work in progress (WIP)
- Enhance Flow :When one task is completed then take next task

Kanban Benefits-

- Flexible in planning
- Incremental delivery of product reduces road block
- Visual Metric
- Shorten cycle time

For transparent workflow management according to requirement each task is first prioritize and then placed in "To-do" list with high priority to low priority. Task on which team is working is moved to the "In-Process" and after completion of task it is moved to the "Done" list.

V. SCRUMBAN

Scrumban is mixture of Scrum and Kanban by combining agility of empiricism with transparent workflow management system. Figure 4 indicates basic mixture of Scrum and Kanban in the form of Scrumban.

Core elements of Scrum that are used in the mixture:

Sprint: Sprint planning, review and retrospective
Pull System: Pull workload into Sprint backlog
Push System: Prioritization and team decision to push work item into Sprint Backlog.

Core elements of Kanban that are used in the mixture:

WIP Limit: Explicit limit indicating how many work items should be in process at a time. Shorten Lead Time: Management and planning of lead time using JIT concept.
Kaizen: Maximize improvement and minimize waste.
CICD: Continuous Integration and Continuous Delivery for the release of completed work items.
WIM: Work item management and its visualization during transition through each stage

Scrumban also can be also used as stepping stone as team seeking to transition from scrum and kanban. For several software development teams, an instantaneous shift to Kanban would be too drastic. Scrumban offers teams the way of learning the way to practice continuous improvement in Kanban without abandoning the familiar structure of Scrum.
Things From Scrum : Iteration, standard, retrospective and demos.

Things from Kanban :

- Wrok in progress limit (WIP)
- Just in time planning

Backlog in Scrumban ;

- Here focus is on reducing the waste i.e. not creating or analyzing too many stories, which are, inform of requirements or defects.
- Deep analysis before starting the development of any story.
- Prioritization when necessary. Provide best thing to work upon

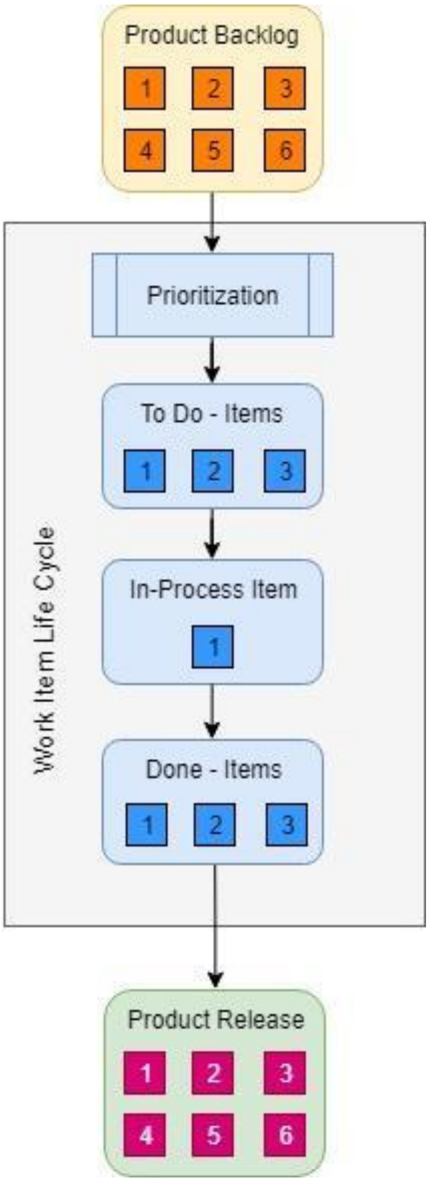

Figure 3. Kanban

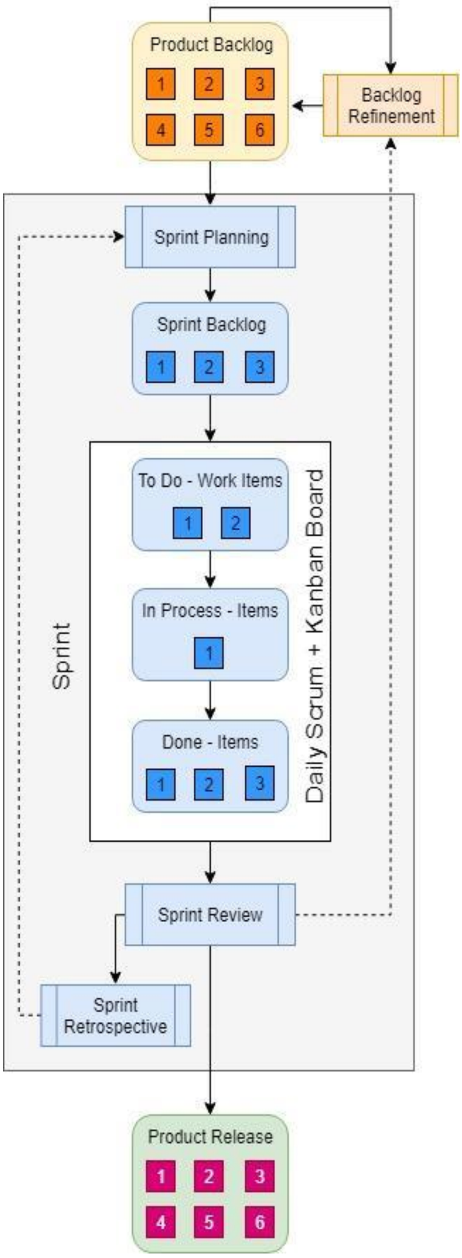

Figure 4. Scrumban flow

VI. Methodology Survey :

a) Time constraints :

As taking time constraints as days, Time required for Scrumban is shorter than the Scrum and kanban

	Days		
Projects	**Scrum**	**Kanban**	**Scrumban**
P1	20	15	7

P2	26	10	8
P3	15	7	5
P4	22	8	5
P5	16	9	9

Table 1. Using timing(days) constraints in different methodologies

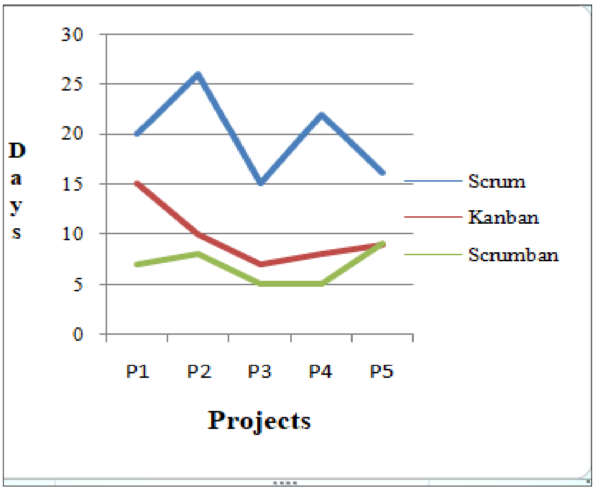

Figure 5. Line chart showing days for different methodology

Scrum is a good framework, care needs to be taken as to not micromanage everything. There should be enough slack time for adhoc task

b) Efforts of each team

Effort can be calculated as following formula : Effort = a(kloc)^b (Using basic cocomo model)
Where KLOC is a measure of the size of a computer program. The size is determined by measuring the number of lines of source code a program has.

Projects	Scrum	Kanban	Scrumban
P1	3390.09	4105.363	3746.913
P2	5189.28	3390.09	3390.09
P3	3390.09	3390.09	3035.05
P4	1295.312	1295.312	1125.856
P5	5189.28	5189.28	3390.09
Total	18454.05	13264.77	14688

Table 2. Efforts calculated using KLOC

By assuming kloc for each project in scrum, kanban and scrumban methology derived efforts for each project
It shows that some of project required less effort using scrumban than kanaban and scrum

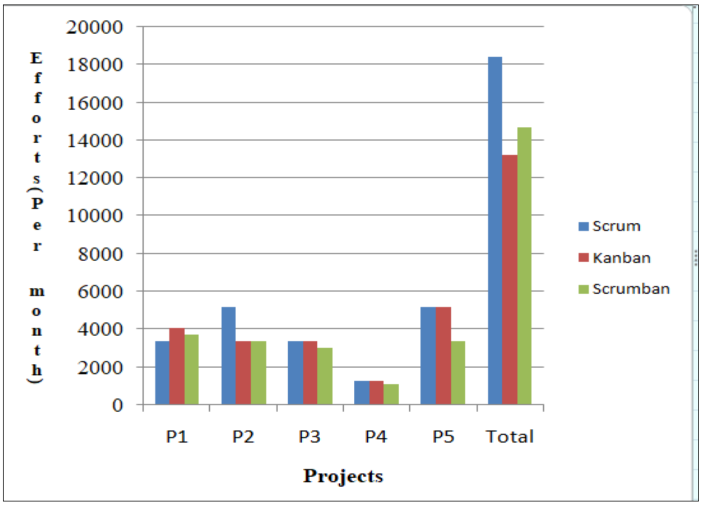

Figure 6. Bar chart showing efforts for different methodology

c) Main tools that you have used or currently using to manage agile processes and activities.
To combine some principles from both the Kanban and Scrum methods into a simple tool (spreadsheet) that we can use both collaboratively and remotely. According to taking survey within Mumbai it seems that 70% industry uses tool like spreadsheet.

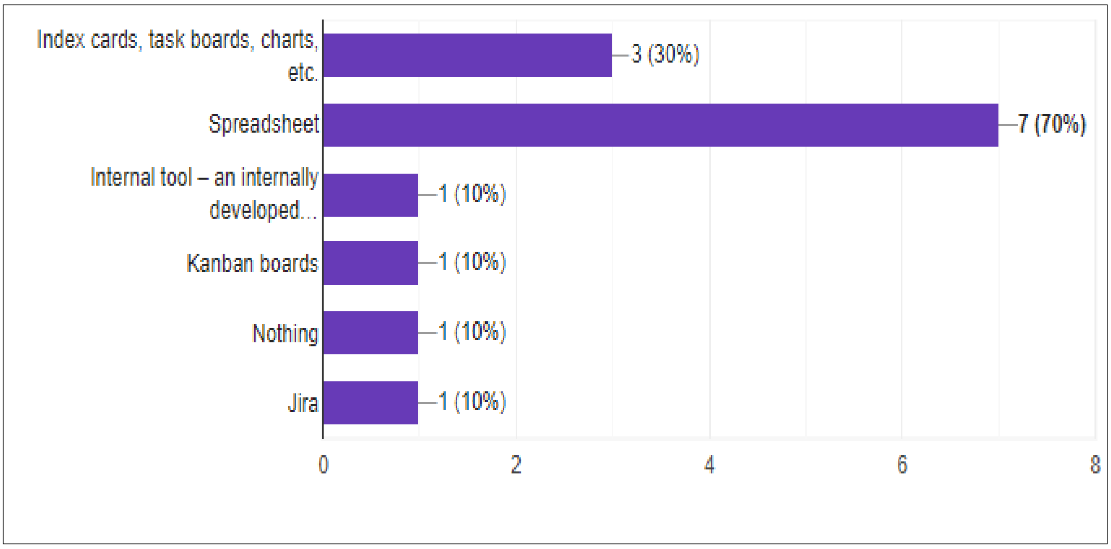

Figure 7. Bar chart showing tools used for different methodology

VII. SOLUTION

1. Can turn into a methodology mish-mash
Solution : Instead of adding more column of " progress "or "doing" we can add row as "blocked", with providing WIP limits.

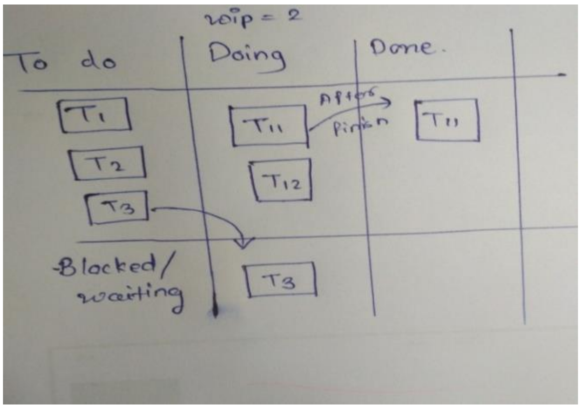

Figure 8. 'flow' back in workflow with WIP limits

2. Hard to track individual team member's effort and contribution
Solution: Voting System and giving credit points. Giving credit based on the following performance factor: Communication, Quantity of Work, Code Quality, Complexity, Dependability, Cooperation

3. Less control for the project manager and no meeting
Solution: Instead of No meeting We can arrange Weekly meeting (max 10 minutes) , that also helps to control over the management

4. Every team-member in Scrumban chooses their tasks themselves
Solution: We must maintain user table that store assigning task and log file that send message is already assign

VIII. ALGORITHM

Task Priority Algorithm

```
STEP 1: RECEIVE input1 (int) FROM (STRING) KEYBOARD for Urgence
STEP 2: RECEIVE input2 (int) FROM (STRING) KEYBOARD for Impact
STEP 3: RECEIVE input3 (int) FROM (STRING) KEYBOARD for Effort
STEP 4: IF input1 VALID OR input2 VALID OR input3 VALID THEN
STEP 5: Allow calculating priority
STEP 6: ELSE
STEP 8: SEND “Undefined” TO SCREEN
STEP 9: END IF
STEP 10: Ask for NEW task priority
STEP 11: CONTINUE from STEP 1
```

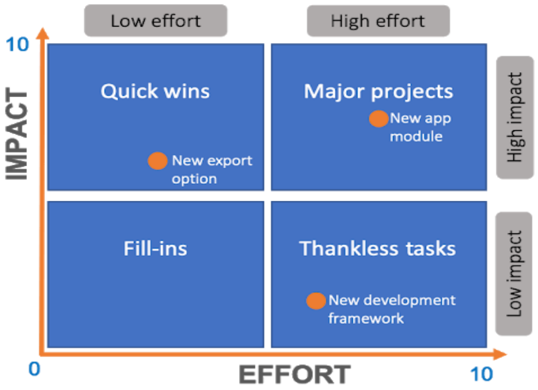

Figure8. Task priority model

IX. CONCLUSION

Different survey based on time constraints, tool that is widely used than the any board, using basic cocomo model to calculate efforts and solution on the selection of Scrum, Kanban and Scrumban were provided to show how Scrumban can be successfully formed based on Scrum and Kanban practices. Thus I have find better solutions to overcome the drawback of scrumban which is inherited by scrum and kanban using task priority matrix, voting system based on performance factor and weekly meeting

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.drdobbs.com/architecture-and-design/the-non-existent-software-crisis-debunki/240165910

[3] Alqudah, M., & Razali, R. (2018). An empirical study of Scrumban formation based on the selection of scrum and Kanban practices. International Journal on Advanced Science, Engineering and Information Technology, 8 (6), 2315-2322.

[4] Patil, Saniav Pandit, and Jitesh R. Neve. "Productivity Improvement of Software Development Process Through Scrumban: A Practitioner's Approach." In 2018 International Conference On Advances in Communication and Computing Technology (ICACCT), pp. 314-318. IEEE, 2018.

[5] Kamini Godbole and Dr. Jitesh Neve, “Kanban: A Definite Answer for Effective Project Execution in Uncertainties,” in Inernational Journal of Advanced Research in Computer and Communication Engineering (IJARCCE), Vol. 6, Issue – 6, pp. 561-565, June 2017.

[6] K. Schwaber and J. Sutherland, "The Kanban Guide for Scrum Team", Scrum.org, 2019

[7] M. Cohn and D. Ford, "Introducing an Agile Process to an Organization", IEEE Computer, vol. 36, issue 6, pp. 74-78, 2003. DOI: 10.1109/MC.2003.1204378

[8] M. Kohlbacher, E. Stelzmann and S. Maierhofer, "Do Agile Software Development Practices Increase Customer Satisfaction in Systems Engineering Projects?", 2011 IEEE International Systems Conference, Montreal, QC, pp. 168-172. 2011, EISBN: 978-1-4244-9493-4, DOI: 10.1109/SYSCON.2011.5929091

[9] N. Nikitina and M. Kajko-Mattsson, "Developer-driven big-bang process transition from Scrum to Kanban", International Conference on Software and Systems Process, 2011, pp. 159-168.