

# Scrum Challenges: An Agile Process Reengineering in Software Engineering

Krunal Bhavsar, Vrutik Shah, Samir Gopalan



**Abstract:** *Business Process Reengineering (BPR) is an organizational strategy seeking identification of critical factors affecting software development practices in Software Engineering Management (SEM). The agile approaches opened the door of BPR in software development organizations. Dynamically changing technological environment has invited software development organizations, towards adoption of reengineering of strategies and practices. Nowadays, Scrum has been trending as an agile framework for Software Engineering Management (SEM) in software development organizations but considerable research work has not been done with Scrum and its successful implementation yet. The goal of this research is to highlight weaknesses and limitations of Scrum like project level documentation, planning, estimation, progress tracker, CI/CD etc.; and identification of challenges and issues with implementation of Scrum in software project development like large scaled project in distributed work environment with multiple Scrum teams, organizational structure, lack of knowledge and awareness about Scrum; using empirical analysis and literature studies. The result of this research will open the door for the next stage of research to overcome the limitation and issues with implementation of Scrum and raise a request of BPR towards focus on improvement in SEM practices in software development organizations by proposing proper training about the empiricism control theory of Scrum that helps organizations in understanding the successful implementation of Scrum and its benefits. This research also raises a question in the form of suggestion that it is good time for a hybrid form of Scrum with other standard agile based frameworks or traditional SDLC methodologies.*

**Keywords:** Scrum, Scrum Challenges, AM - Agile Methodology, BPR - Business Process Reengineering, ESE – Empirical Software Engineering, SPI – Software Process Improvement, SEM – Software Engineering Management, SE – Software Engineering

## I. INTRODUCTION

Scrum has been trending in software development organizations [10] as a standard agile based framework for software project development and management that involves incremental and iterative process cycle for rapid delivery of functional product under the roof of agile and on the base of empiricism for Software Engineering Management (SEM).

Revised Manuscript Received on February 28, 2020.

\* Correspondence Author

**Krunal Bhavsar\***, Research Scholar, Computer Science & Engineering, Indus University, Ahmedabad, India. Email: krunalbhavsar@engineer.com; krunalbhavsar.rs@indusuni.ac.in. Contact: +91-9737007007.

**Dr. Vrutik Shah**, Research Guide, Computer Science & Engineering, Indus University, Ahmedabad, India.

**Dr. Samir Gopalan**, Research Co-Guide, Business Administration & Management, Indus University, Ahmedabad, India.

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an open access article under the CC-BY-NC-ND license <http://creativecommons.org/licenses/by-nc-nd/4.0/>

Software Engineering (SE) is a set of functional procedure that must be executed in a proper sequence to achieve desired goal. Technological world has been consistently emerging BPR request in the software domains and the software development organization should be ready to adopt such new technologies [6]. Agile methodologies approach, no or minimum documentation for the software project development by relying on software development team to manage it but the software project development and management processes become more critical as the size of project grows [35]. Project estimation for additional cost, resource, infrastructure etc., becomes more complex [36]. It becomes typical for the organization to control the projects with unclear goal and frequently changing demands by stakeholders. Scrum is very suitable for such type of projects but it becomes difficult for Product Owner and Scrum Master to manage such projects when projects run into indefinite timeline and cost. Implementation of Scrum does not provide mechanism for project scheduling and estimation control to the top management. Ballpark estimation for software product development, leads towards unclear vision, goals and directions. Time and cost estimation are prime factors for the organizations [37], when they are thinking about software product development. Unclear estimation invites uncertain risks in project. Some of the organizations do not allow self-organization of development team. They prefer a supervisor like technical project manager or a team leader of a team for organizing, monitoring and controlling development activities. It's time to highlight challenges and limitations of Scrum for the researchers to focus and step into suggestive resolution, as such arguments have raised questions against ideology of Scrum and its persistence in future of software engineering. Krunal et al (authors) have stepped into this research with goal of ABPR (Agile Business Process Reengineering) in Scrum and its terminology; and feasibility of integration of automation technologies [12], [38] to empower Scrum capabilities.

## II. RESEARCH OBJECTIVES AND METHODOLOGY

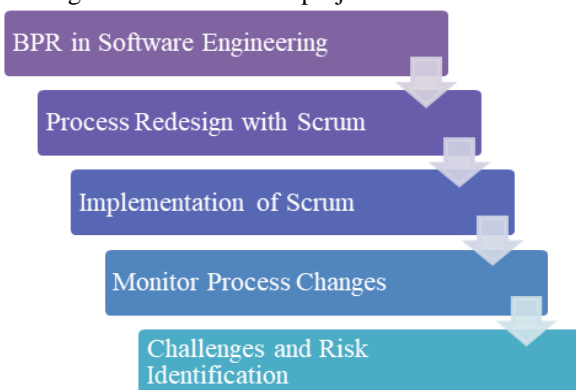
This research represents Scrum as a trending agile framework for SEM but still there are some challenges for the Scrum. The empirical analysis of Scrum rules and regulations and its implementation based on empirical study, is the approach of this research. The aim of the research study highlights the challenges with implementation of Scrum in SEM (Software Engineering Management) for the organizations looking for the BPR in their software development units with the help of Scrum, using empirical analysis and reviews.

Following are primary objectives of the research:

- Highlight the issues with implementation Scrum
- An empirical analysis of limitations of Scrum
- Proposed resolution against challenges of Scrum by raising a request for further research on scope of improvement in Scrum.

### III. SCRUM AND BPR

Scrum is a Software Engineering Management (SEM) framework built on empiricism, under the shadow of agile principles [7]. The keyword ‘Scrum’ was shouted for the first time in 1986 [8] as an agile approach, which transformed strategies for software development businesses. Schwaber and Sutherland announced Scrum as a standard agile software development framework in 1993 [9], which has been proven as a creative and productive approach for software development projects. Scrum has been fascinating Software Engineering Management (SEM) framework as BPR approach in the software development organizations since past few years due to some of the empiricism control theory based characteristics of Scrum like artifacts, roles, values, events, pillars [10]. Mellor et al [11] introduced agile manifesto and its principles over the traditional software development practice which strengthened competencies of organization to control internal factors and successfully drove BPR with the help of Scrum. PLCF (Process Life Cycle Framework) [2] is a life cycle model of BPR for software development organizations. Integration of Scrum with PLCF improves capabilities of organization towards adoptions of BPR in SEM. Existing project management tools are demanding integration of extended Scrum features with the help of AI (Artificial Intelligence) [14] based ML (Machine Learning) [12] technologies algorithms like k-means data clustering integration with Scrum tree [13]. Adoption of AI in SPM (Software Project Management) is increasing [14] for accurate and automated planning and scheduling of the estimation of project.



**Fig. 1. BPR in SE using Scrum**

Figure 1 represents the BPR life cycle stages for implantation of Scrum in software engineering and identification of risk and challenges as a goal of this research. Following are steps of BPR in SEM:

- Raise a request for need of BPR in Software Engineering
- Analyze and redesign processes with the help of Scrum
- Implement Scrum as an agile BPR approach in Software Engineering (SE) practices
- Monitor process changes and document feedbacks

- Identify Risk and challenges with Scrum

### IV. LIMITATIONS OF SCRUM

According to ‘The Kanban Guide for Scrum Team’ [5], Scrum does not provide implicit or explicit policies about how to accomplish explicit visibility of work flow into PB (Product Backlog), SB (Sprint Backlog) and achievement of DoD. It does not specify the internal states of each workflow item passing through each state during the Sprint. Software development organizations and Scrum teams in the organization, follow their own strategy for workflow management to achieve their goal and it leads towards inconsistencies and results into decrease in productivity and delay in product release. Following are considerable limitations of Scrum with respect to the nature of real world software development environment, which need to be discussed and focused.

#### A. Question - How?

Scrum provides limited guidance about how to accomplish specific task while adopting Scrum. It always raises questions:

- How to start and end?
- How to pile up and manage the backlog items?
- How work item will flow from external environment to PB, from PB to SB and SB to DoD through internal states of work item in Sprint?
- How the involvement of none Scrum team members (like client or management of organization or operational team member) should be taken care by the Product Owner (PO)?
- How to design/define the team structure having cross functional expertise, which is necessary to accomplish the Sprint goal, when there are possibilities of having different kind of features need direct involvement of the stakeholder as a Scrum team member for a specific Sprint?
- How to manage PB when multiple Scrum teams are working on same Product?

#### B. External Stakeholder

Scrum strictly prohibits direct involvement of external stakeholder, is the main weakness of Scrum. Transparency of product and suggestive inputs from client are essential for success of the product development. Inspection by client clears the direction of product for development, so the final product will be significantly matching with initial vision of product. In 2001, Cockburn and Highsmith highlighted [1] a fact about potential weaknesses during their research study on Scrum that when the team is building product for the external stakeholder (client), he or she must be involved into Sprint review for inspection, acceptance and feedbacks on deliverable product increment.

#### C. Team Size

Scrum limits team size between 3-9 members per Sprint. Team with the size of 3-9 members, is the best team as per empiricism control theory,

to avoid waste of time for unnecessary discussion and engagement of resources but how to get the work done for a large size project with a single team having 3-9 team members? Large projects require more team members than the Scrum limits. Also, 15 minutes of duration for Daily Scrum meeting will not be enough if team is large.

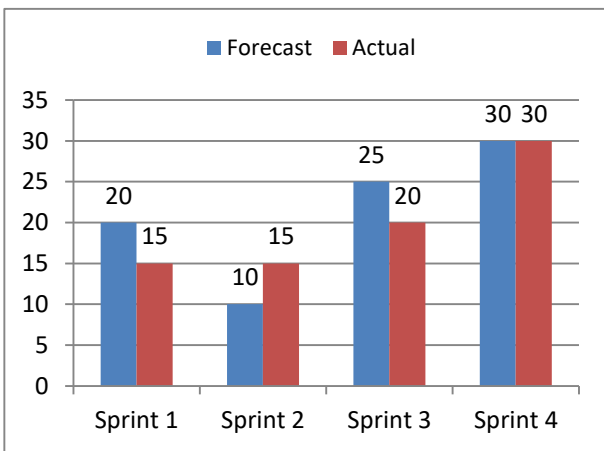
**D. Progress Report**

Process and progress report trackers defined in Scrum rulebook provides limited information on current state of Sprint and does not provide clear vision towards achievement of DoD.

Velocity chart represents progress of work with respect to User Story Points or Epic or Teams or Sprint or Product Backlog against comparison with timeline to accomplish that task.

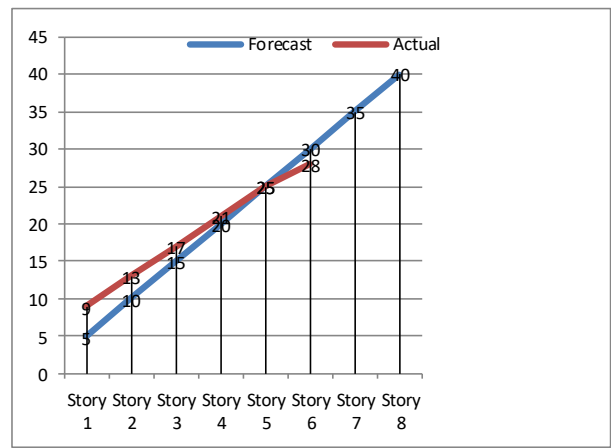
- Empiricism based estimations in Scrum are always misleading as User Story points are subjective statements, which may further divisible in various tasks with different level of complexities.
- Velocity can't provide accurate estimation about completion of task pending in pipeline and capacity.
- Expected velocity is dependent on current velocity of team, which may provide inaccurate information.
- Velocity ignores QA (Quality Analysis) test changes and technical debts.

Figure 2 demonstrates Scrum velocity chart with a sample Sprints in X-axis and no. of user stories Y-axis with actual and forecasted values for a software product development.



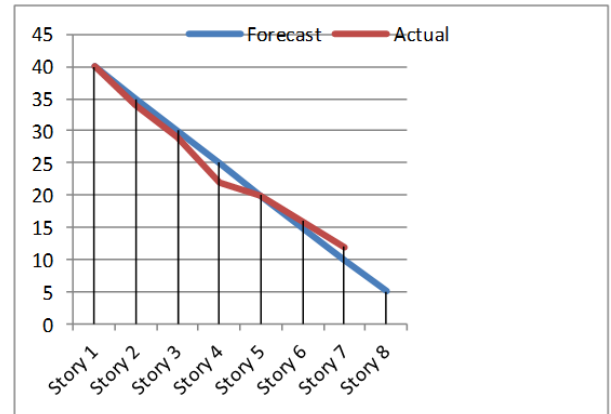
**Fig. 2.Scrum Velocity Chart Demonstration**

While burn up chart is used for the estimation of actual pending task towards the Sprint goal and also having the same issues that the velocity chart like approximate estimation does not provide accurate time line for all pending tasks. Figure 3 demonstrates Scrum burn up chart with a sample user stories in X-axis and no. of user stories Y-axis with actual and forecasted values for a software product development.



**Fig. 3.Scrum Burn-Up Chart Demonstration**

Burn down chart is used for the estimation of actual completed task (red line), with comparison to estimation (blue line) - towards the Sprint goal.



**Fig. 4.Scrum Burn-Down Chart Demonstration**

Figure 4 demonstrates Scrum burn down chart with a sample user stories in X-axis and no. of user stories Y-axis with actual and forecasted values for a software product development.

**V. CHALLENGES WITH SCRUM**

Software development organizations have been dealing with following challenges for successful implementation of Scrum:

**A. Resistance**

Resistance to change management in organization who transforms their software development practices from traditional to agile approach.

- Resistance by external stakeholders or client, when they are not aware about Scrum and its benefits.
- Resistance by management of organization, when they feel less anxious and may not understand the benefits to organization due to changes made by Scrum.
- Resistance by team members,
  - If team members are not ready to adopt Scrum in their software engineering practices.

## Scrum Challenges: An Agile Process Reengineering in Software Engineering

- If the development team members are not aware about Scrum and its benefits
- If the development team is failed for successful implementation of Scrum in development activities.
- If the team members are not mature enough to become a part of Scrum team due to lack of expertise as a cross functional team.
- If the team members are not ready to take a responsibility or accountability of self-organization.
- When product owner can't (or not ready) control Sprint inflow or unable to handle interruption of top management.

### B. Large Scale of Product

Scrum is best suited for the project where single team is enough to provide expectable deliverables within available timeline, whether product large, medium or small. But when overall product development time line is shorter for a single team to generate outflow, an organization needs to involve multiple Scrum teams in single product development. At this time there are possibilities of issues to be risen.

- Vision of final product may become unclear, as Scrum is iterative and incremental model and often accepts amendment in initial product vision.
- It is almost impossible to provide accurate estimation of development work to client prior to development begins.
- Roles of Scrum team members may be limited as large product development requires direct involvement of internal and external stakeholders also.
- Multiple teams may require for development, which directly leads toward issues with collaboration, accountability, responsibility and self-organization in development team.

### C. Multiple Teams

Scrum insists that team of 3-9 members is enough to accomplish all the development work required for the software product but for the large scale software product, we may need multiple teams for a single product.

- Single product owner is required even if there are multiple teams. Product owner has to extend his capacity to provide necessary support to all team members.
- Coordination issue between cross team members for a Scrum master, if there is one Scrum Master for all the teams.
- Product backlog management issue, if Sprint duration for teams is not consistent.
- Developed work is required to be merged with in same product which may conflict, when there is inter team dependencies.
- Failure (cancellation of Sprint) of one team may effect on other teams also.

### D. Lack of Knowledge and Training

As a Scrum team member, it is very essential for us to understand the rules and regulations of Scrum before implementing in software development practice.

- Most of the team members are unaware about Scrum process knowledge and not trained well.

- Scrum team member is not an individual in a Scrum team, who is accountable for their own task. All team members are responsible for the resolution of any impediment occurs during the Sprint.
- Scrum team members should be mature enough to accomplish their responsibility in shared and distributed environment.

### E. Distributed (Remote) Environment

Collaboration between Scrum team members is prime issue when team members are located in (remote) distributed environment.

- Physical presence (face to face communication) of team members in Daily Scrum is almost impossible.
- Scrum Master's job becomes more typical and lot of indirect coordination is required to communicate with team members.
- Distributed teams are more dependent on documentation due to lack of face to face communication and team members more often try to handoff responsibilities.

### F. Role of Team Members

Scrum has limited roles in a team, which are: Product Owner, Scrum Master and Development Team.

- Product owner is accountable for defining customer value in a product.
- Scrum Master is responsible for ensuring implementation of Scrum.
- Development is responsible for resolution of all of the issues in between development team and activities as a self-organizing team.

As per standard SDLC protocols, various types of stakeholders should be the part of the Scrum team and should directly involve into software development for the success of a product:

- Requirement gathering analysis should be accomplished by business analyst team in collaboration with client and its business.
- High level of planning and scheduling should be prepared by Product and project managers.
- System design should be created by technical architects for each technology.
- Execution is responsibility of development team.
- Testing should be done by QC team and then client also just prior to production release of product.
- Project delivery should be maintained by DevOps team.

According to Scrum, all of (above) these operations should be handled by Scrum team roles (members) only. So, it indirectly indicates that Scrum team should be cross functional with all expertise which are essential to accomplish all tasks (operations) and that must be incorporated in PBI (Product Backlog Item).

### G. Estimation

No visibility or very low visibility of the product vision estimation is the biggest issue with estimation in Scrum. Scrum allows estimation of work during Sprint Planning for the items in Sprint Backlog (SB) only.

There is no provision about full product development estimation in Scrum. This will be major hindrance for software development organization while locking the product estimation agreement with client.

## VI. EMPIRICAL ANALYSIS

Some of the issues with the implementation of Agile Scrum, like involvement of management and external stakeholders into team etc. were determined by Livermore during his survey in 2008 [3]. During the same time, Cho also determined [4] factors impacting implementation of Scrum are: unawareness about benefits of Scrum, lack of knowledge and incomplete training, interruption by top management or external stakeholders, unclear vision about product goal and backlog management. Raza and Majeed exposed [22] some issue with Scrum using an empirical survey in 2012 that improper training about Scrum resulted to lack of knowledge and improper implementation of Scrum in the organizations. Due to limitations of Scrum like condemn rapid updates in product vision, tendency of over budget, weak documentation and poor performance for a large scale project, Ahmad et al [32] proposed that hybrid combination of agile frameworks in 2014. While in 2015, Anand and Dinakaran [23] systematically studied literature reviews on technical issues with agile Scrum like unclear guidance about prioritization of Product Backlog (PB) and Sprint Backlog (SB) work items, failure of implementation of Scrum in distributed environment, no existence of Scrum's ideal development team and suitability of organizational culture. In 2016, Agrawala et al concluded [19] that due to up-front planning limitations, agile methodologies have several issues like budget estimation, documentation, product vision and unsuitable for small organizations. In 2016, Alqudah and Razali [30] reviewed literatures on implementation of agile methodologies like Scrum, XP (Extreme Programming), DSDM (Dynamic System Development Method), CC (Crystal Clear) in the large scaled projects in large organizations where they observed limitations of Agile methodologies in fulfilling requirement of large scaled project like size of team members, communication and collaboration between them, involvement of customers. They also recommended that scaled Agile methodologies can only overcome such limitations. In 2017, Salah et al [31] argued that delay in project delivery, over budget and failure in achieving DoD in Scrum are still persist in software project development and management; which indicates ABPR (Agile Business Process Reengineering) for Scrum. Ashraf and Aftab argued [33] that Scrum should be integrated with other spices of Agile framework to improve productivity of outcome, using systematic mapping study in 2017. Miler and Kajdy evaluated during their workshops survey in 2018 [18] that there is still need to modify design of Scrum framework, primarily role of Scrum Master. Mishra and Abdalhamid concluded [25] various quality related issues with Scrum as a result of their research work in 2018. They found that software product and project quality related research were paid more attention that discovered product quality issues like poor visibility of product, low ratio of quality acceptance and effectiveness of quality process. In 2018, Ma'arif et al [29] observed challenges like differences in

behavior of technological needs, scare in team members about adoption of Scrum, involvement of external stakeholders; for implementation of Agile Scrum in Information System projects like ERP, ICT services and enterprise architectures in the region of Malaysia; by conducting interviews with the ITSS domain experts. In 2018, Hanslo and Mnkandla [34] analyzed challenges with Scrum like lack of expertise because of limited roles in a team, suitability of organizational culture with Scrum and poor communication; using narrative reviews.

In 2019 [16], Ganesh and Narayanan implemented Scrum in ERP (Enterprise Resource Planning) phase development and analyzed that Scrum has helped in software quality delivery but they also observed challenges like involvement of team members and product owner along with external stakeholders, and collaboration with each other. Software Engineering Management requires proper effort estimation techniques. Effort estimation of entire project is the part of project planning and scheduling phase of SDLC and essential process prior to system design and development activities. Traditional SDLCs allow complete estimation of project for the confirmation of internal and external stakeholders. In 2019, Bhaskaran and Jayaraj observed [17] that agile methodologies do not have such perception. Estimation process comes into consideration during the Sprint planning only. It is difficult to estimate software product development effort using Scrum. Ram and Vijaykumar [26] surveyed the philosophy of agile team members to measure the level of key aspects: Knowledge Sharing, Collaboration and Trust in software development organization, in 2019. Mean score of survey represented that level of Trust in development team about management decision, was very lower in comparison of other key aspects. It proves that development team just follows management decision. They do not contribute towards improvements as management failed in playing pure Scrum role in a Scrum team. In 2019, Mkpjojiogu et al surveyed [21] in a software development startup organization of Saudi Arabia, to measure the negative impact of adoption of agile Scrum in startup. Along with some positive motivations, they found negative impact also, which was lack of knowledge and

awareness about proper implementation of Agile in team members. Arbain et al [27] conducted an empirical case study in 2019, to explore impact of NFR (Non-Functional Requirement) change in Agile software approach and they evaluated that agile methodology is dependent on the necessity of traceability which provides a better option for NFR (Non-Functional Requirement) change.

Rai et al [28] performed a research study on agile software project development methodologies like Scrum and XP and identified several risk elements with use of Agile in software development, which are distributed environment, fixed resources, no technical debt build up, automation testing, project planning, Agile training, team members and their roles.

## VII. RESULTS AND RECOMMENDATIONS

Scrum is very popular in software development organizations as an agile Software Engineering practices framework, although we have carried out several challenges due to limitations of Scrum and recommendations for improvements in successful implementation of Scrum and its standard practices.

### A. Challenges and Limitations

Scrum very complex to master. We have determined following challenges and limitations, of Scrum represented in table-I with brief explanation for each:

**Table I: Limitations of Scrum**

Parameter	Limitations
Guidance	Limited guidance from the Scrum community.
External Stakeholder	Restricted external stakeholders.
Team Size	Restricted team size.
Role	Limited roles in a Scrum team
Project Tracker	Progression view at project level.
BPR	Resistance to change management.
Estimation	Full product development estimation is not feasible.
Product Vision	Unclear product vision.
Unawareness	Lack of awareness Scrum.
Distributed Environment	Collaboration issue.
Skill and Expertise	Unavailability specific skilled resource.
Documentation	No or minimum documentation.

- Explanation about each parameter of limitations of Scrum as per Table I:
  - Guidance: Limited guidance from the Scrum community about how to successfully implement Scrum in software engineering practices.
  - External Stakeholder: Restricted direct involvement of external stakeholder like client in Scrum team unless he or she is ready take accountability as a Product Owner (PO).
  - Team Size: Restricted team size between 3 and 9 is limited for large scale software product development.
  - Role: Restricted roles of Scrum team member like Product Owner, Scrum Master and Development team member are limited for large scale software product development where involvement of additional role is essential as per expertise.
  - Project Tracker: Process progress reports like burn-down/burn-up and velocity charts are designed based on empirical estimation process that does not provide clear vision for achievement of full product goal, during the Sprint.
  - BPR: Resistance to change management for the implementation or execution of Scrum, by external (client or consultancy providing support to organization) or internal stakeholder (management team or operational team or Scrum team member) of organization.
  - Estimation: Complete product development estimation

with required level of accuracy is almost not feasible at initial stage of product, prior to development activities begin.

- Product Vision: Vision of the product has found unclear for a large scale product as Scrum is about continuous iteration and increment, product vision changes often.
- Unawareness: Lack of awareness and knowledge about Scrum, its benefits and implementation.
- Distributed Environment: Lack of collaboration in distributed (remote) team environment.
- Skill and Expertise: Unavailability of requirement specific expertise/skill in a team due limited roles.
- Documentation: Scrum agile principles believe in no SDLC documentation for the complete project or product. It becomes critical for Scrum team to plan and schedule project development.

### B. Recommendations

The result argues that Scrum is trending as an agile software engineering management framework [10] although there are some limitations and challenges for Scrum in software development organizations. Our recommendations for BPR in Scrum practices with regards to its current challenges and its successful implementation are represented in table II with brief explanation for each:

**Table II: Recommendations**

Parameter	Recommendation
Training	Training by Professional Scrum Trainer (PST).
External Stakeholder	Direct member of Scrum team.
Role	Traditional Role
Documentation	Tradition Project Documentation
Agile Integration	Hybrid Framework

- Explanation about each parameter of recommendations as per Table II:
  - Training: Proper guidance and training about Scrum, its benefits and its successful implementation should be provided to stakeholders and Scrum team members by the Professional Scrum Trainer (PST).
  - External Stakeholder: External stakeholder should be direct member of Scrum team, who can inspect and discuss with Product Owner about development progress and changes whenever it is required.
  - Traditional Role: Involvement of traditional roles like Business Analysts, Operational Manager, Quality Analyst and Controller are essential as their skill specific requirement cannot be ignored or replaced in the Scrum.
  - Documentation: Just like a traditional project documentation of the project with specifications about requirement analysis and project estimation is necessary to define the clear goal of project and business deal with client.
  - Agile Integration: We strongly recommend the integration of any other agile based framework with Scrum; not as a replacement of Scrum rules but as an amendment into the rules of Scrum.

The issue related to knowledge and awareness of Scrum is controllable by software development organizations and can be resolved by providing proper training about Scrum and its successful implementation. But the uncontrollable challenges for organizations that have arisen from the root of Scrum need further investigation for their resolution either by correction or improvement or amendment with existing Scrum rules or integration with any other agile framework.

### VIII. CONCLUSION

Today's dynamically changing technological world, has forced software development organizations towards BPR. Scrum has raised a request of BPR in software development organizations with its characteristics and artifacts which leads remarkable improvements in SEM (Software Engineering Management) practices and deliverable results. But this research also carried out as a result of this research that there are several limitations of Scrum that has been challenging for its implementation as a standard approach for software development and management practices for software development organizations, which are: "Question - How to accomplish and achieve goal with the help of Scrum?", involvement of external stakeholders, team size, progress report mechanism, resistance by organization or stakeholders or team members, deal with large scale of software product development, lack of training and awareness about successful implementation of Scrum, multiple team in distributed environment and their roles and most critical challenge is inaccurate estimation of complete product development during initial stage when project development agreement related activities are confirmed with client.

This research has also concluded that internal factors and controllable issues can be resolved by software development organizations can be resolved. But the external factors and uncontrollable challenges for organizations have raised a request for an agile integration in the form of hybrid frameworks which extends the capabilities of agile based frameworks to overcome existing uncontrollable challenges also.

### IX. FUTURE ENHANCEMENT

It's good time to jump into the Scrum and discover the ways which can improve the existing practices of Scrum and uncontrollable challenges in the software development organizations. Furthermore it's for an answer to the question: "Does the Scrum really need integration with any other standard agile approach in the form of a hybrid framework that enables the scope of BPR in SEM?" This question opens the door for next stage of research on agile integration of Scrum and Kanban [20] with traditional SDLC like Waterfall [24] as conceptual hybrid form of Software Engineering Framework (SEF). It's also time to explore the scope of Artificial Intelligence (AI) [14] and Machine Learning (ML) [12] in Software Engineering Management (SEM) frameworks.

### ACKNOWLEDGEMENT

As a part of my academic affiliation with Indus University for the doctorate degree program in Computer Science and Engineering, through this empirical analysis we have

discovered some of the challenges for implementation of Scrum in software development organization. A comparative research about Scrum with other standard Agile based frameworks and resolution of Scrum challenges using hybrid form of frameworks, will be our next goal.

### ABBREVIATIONS

- A** ABPR - Agile Business Process Reengineering  
AI – Artificial Intelligence  
AM – Agile Methodology
- B** BPR – Business Process Reengineering
- D** DoD – Definition of Done
- E** ERP - Enterprise Resource Planning
- I** ICT - Information Communication Technology  
ITSS - Information Technology Security System
- M** ML – Machine Learning
- P** PB – Product Backlog  
PB – Product Backlog Item  
PLCF – Process Life cycle Framework  
PO - Product Owner  
PST - Professional Scrum Trainer
- Q** QA - Quality Analysis/Analyst  
QC - Quality Control
- S** SB – Sprint Backlog  
SDLC – Software Development Life Cycle  
SE – Software Engineering  
SEF – Software Engineering Framework  
SEM – Software Engineering Management  
SPM - Software Project Management

### REFERENCES

1. J. Highsmith and A. Cockburn, "Agile Software Development: The Business of Innovation", IEEE Computer, vol. 34, issue 9, pp. 120-122. DOI:10.1109/2.947100
2. K. Bhavsar, V. Shah and S. Gopalan, "Process Life Cycle Framework: A Conceptual Model and Literature Study of Business Process Re-Engineering for Software Engineering Management", CiiT International Journal of Software Engineering and Technology, vol. 11, issue 6, pp. 096-100, 2019. ISSN: 0974-9748
3. J. Livermore, "Factors that Significantly Impact the Implementation of an Agile Software Development Methodology", Journal of Software, vol. 3, issue 4, pp. 31-36, 2008.
4. J. Cho, "Issues and Challenges of Agile Software Development with Scrum". Issues in Information Systems (IACIS - International Association for Computer Information Systems, Stillwater, OK., USA), vol. 9, issue 2, pp. 188-195, 2008.
5. K. Schwaber and J. Sutherland, "The Kanban Guide for Scrum Team", Scrum.org, 2019
6. S. Nerur, R. Mahapatra and G. Mangalaraj, "Challenges of migrating to agile methodologies", Communications of the ACM, vol. 5, issue 5, pp. 73-78, 2005.
7. K. Schwaber and J. Sutherland, "The Scrum Guide", Scrum.org, 2017.
8. H. Takeuchi and I. Nonaka, "The New Product Development Game", Harvard Business Review, 1986.
9. J. Sutherland and K. Schwaber, "The Scrum Papers: Nut, Bolts, and Origins of an Agile Framework", Scrum Inc., draft: 29 Jan 2011, Paris.
10. K. Bhavsar, V. Shah and S. Gopalan, "Scrum: An Agile Process Reengineering in Software Engineering", International Journal of Innovative Technology and Exploring Engineering (IJITEE), vol. 9, issue 3, pp. 840-848, 2019, ISSN: 2278-3075.
11. S. Mellor, R. C. Martin, J. Kern, K. Schwaber, J. Grenning, W. Cunningham, M. Beedle, A. Bennekum, K. Beck, A. Cockburn, M. Fowler, J. Highsmith, A. Hunt, R. Jeffries, B. Marick, J. Sutherland and D. Thomas, "Manifesto for Agile Software Development", AgileManifesto.org, 2001

# Scrum Challenges: An Agile Process Reengineering in Software Engineering

12. K. Bhavsar, V. Shah and S. Gopalan, "Machine Learning: A Software Process Reengineering in Software Development Organization", *International Journal of Engineering and Advanced Technology (IJTEAT)*, vol. 9, issue 2, pp. 4492-4500, 2019. ISSN: 2249-8958, DOI: 10.35940/ijteat.B4563.129219
13. R. A. Mahajan, S. K. Yadav and S. A. Mahajan, "Development and Integration of Scrum Tree Algorithm with K-means Data Clustering", *International Journal of Engineering and Advanced Technology (IJTEAT)*, vol. 8 issue 6, 2019, pp. 4248-4251, ISSN: 2249 – 8958
14. K. Bhavsar, V. Shah and S. Gopalan, "Business Process Reengineering: A Scope of Automation in Software Project Management using Artificial Intelligence", *International Journal of Engineering and Advanced Technology (IJTEAT)*, vol. 9, issue 2, pp. 3589-3594, 2019. ISSN: 2249-8958, DOI: 10.35940/ijteat.B2640.129219
15. S. Al-Ratrouf, "Practical Implementation of Agile Approaches in Teaching process", *International Journal of Engineering and Advanced Technology (IJTEAT)*, vol. 8, issue-4, 2019. ISSN: 2249 – 8958
16. R. C. Narayanan and N. Ganesh, "Challenges Faced in the Enterprise Resource Planning Material Management Section when Transitioning Towards Agile Software Development", *International Journal of Engineering and Advanced Technology (IJTEAT)*, vol. 8, issue 6, pp. 3472-3475, 2019. ISSN: 2249–8958
17. V. Jayaraj and N. A. Bhaskaran, "A Hybrid Effort Estimation Technique for Agile Software Development (HEETAD)", *International Journal of Engineering and Advanced Technology (IJTEAT)*, vol. 9, issue 1, pp. 1078-1087, 2019, ISSN: 2249–8958
18. J. Miler and K. Kajdy, "Problems and Solutions of Software Design in Scrum Projects", *Proceedings of the Federated Conference on Computer Science and Information Systems (IEEE)*, vol. 15, pp. 975–978, ISSN 2300-5963, DOI: 10.15439/2018F151
19. A. Agrawala, M. A. Atiq and L. S. Mauryac, "A Current Study on the Limitations of Agile Methods in Industry Using Secure Google Forms", *International Conference on Information Security & Privacy (ICISP2015)*, vol. 78, pp. 291–297, 2016
20. K. Bhavsar, V. Shah and S. Gopalan, "Scrumban: An Agile Integration of Scrum and Kanban in Software Engineering", *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, vol. 9, issue 4, 2020. ISSN: 2278-3075.
21. E. O. C. Mkpojiogu, N. LailyHashim, A. Al-sakkaf and A. Hussain, "Software Startups: Motivations for Agile Adoption", *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, vol. 8, issue 8S, pp. 454-459, 2019, ISSN: 2278-3075.
22. A. Raza and H. Majeed, "Issues and Challenges in Scrum Implementation", *International Journal of Scientific & Engineering Research*, vol. 3, issue 8, pp. 1-5 2012, ISSN 2229-5518
23. R. V. Anand and M. Dinakaran, "Issues in Scrum Agile Development Principles and Practices in Software Development", *Indian Journal of Science and Technology*, vol. 8, issue 35, 2015, ISSN: 0974-5645, DOI: 10.17485/ijst/2015/v8i35/79037
24. K. Bhavsar, V. Shah and S. Gopalan, "Scrumbanfall: An Agile Integration of Scrum and Kanban with Waterfall in Software Engineering", *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, vol. 9, issue 4, 2020. ISSN: 2278-3075.
25. D. Mishra and S. Abdalhamid, "Software Quality Issues in Scrum: A Systematic Mapping", *Journal of Universal Computer Science*, vol. 24, issue 12, pp. 1690-1716, 2018.
26. V. Ram and T. Vijayakumar, "Effects of Agile adoption on Trust, Knowledge Sharing and Collaboration in IT Organizations", *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, vol. 8, issue 12, 2019, pp. 2652-2655, ISSN: 2278-3075. DOI: 10.35940/ijitee.K2219.1081219
27. A. F. B. Arbain, D. N. A. Jawawi, W. M. N. B. W. Kadir and I. Ghani, "Requirement Traceability Model for Agile Development: Results from Empirical Studies", *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, vol. 8, issue 8S, pp. 402-405, 2019. ISSN: 2278-3075
28. A. K. Rai, S. Agarwal, M. Khaliq, A. Kumar, "Quantitative Analysis of Development Environment Risk for Agile Software through Machine Learning", *International Journal of Recent Technology and Engineering (IJRTE)*, vol. 7, issue 6, pp. 83-89, 2019. ISSN: 2277-3878
29. M. Y. Ma'arif, S. M. Shahar, M. F. H. Yusof and N. S. M. Satar, "The Challenges of Implementing Agile Scrum in Information Systems Project", *Journal of Advanced Research in Dynamical and Control Systems*, vol. 10, issue 9, pp. 2357-2363, 2018. ISSN: 1943-023X
30. M. Alqudah and R. Razali, "A Review of Scaling Agile Methods in Large Software Development", *International Journal on Advanced Science, Engineering and Information Technology (IJASEIT)*, vol. 6, issue 6, pp. 828-837, 2016. ISSN: 2088-5334, DOI: 10.18517/ijaseit.6.6.1374
31. A. Salah, N. R. Darwish and H. A. Hefny, "Towards a Hybrid Approach for Software Project Management using Ontology Alignment", *International Journal of Computer Applications (IJCA)*, vol. 168, issue 6, pp. 12-19, 2017. ISSN: 0975 – 8887
32. G. Ahmad, T. R. Soomro and M. N. Brohi, "XSR: Novel Hybrid Software Development Model (Integrating XP, Scrum & RUP)", *International Journal of Soft Computing and Engineering (IJSCE)*, vol. 2, issue 3, pp. 126-130, 2014. ISSN: 2231-2307
33. S. Ashraf and S. Aftab, "Scrum with the Spices of Agile Family: A Systematic Mapping", *International Journal of Modern Education and Computer Science (IJMECS)*, vol. 9, issue 11, pp. 58-72, 2017. ISSN: 2075-017X, DOI: 10.5815/ijmeecs.2017.11.07
34. R. Hanslo and P. E. Mnkandla, "Scrum Adoption Challenges Detection Model: SACDM", 2018 Federated Conference on Computer Science and Information Systems (FedCSIS), Poznan, pp. 949-957, 2018.
35. D. Suresh and A. Sivakumar, "Impact of Schedule Management Plan on Project Management Effectiveness", *International Journal of Engineering and Advanced Technology (IJTEAT)*, vol. 9, issue 1, pp. 3350-3355, 2019. ISSN: 2249–8958, DOI: 10.35940/ijteat.A1515.109119.
36. D. Sivakumar and C. Sureshkumar, "Time of Development and Fitness Analysis of Modified COCOMOII Model for Software Projects", *International Journal of Recent Technology and Engineering (IJRTE)*, vol. 8, issue 2, 2019. ISSN: 2277-3878, DOI: 10.35940/ijrte.B3304.078219.
37. K. S. N. Prasad and M. V. V. Saradhi, "Comprehensive Project Management Framework using Machine Learning", *International Journal of Recent Technology and Engineering (IJRTE)*, vol. 8, issue 2S3, pp. 1373-1377, 2019. ISSN: 2277-3878, DOI: 10.35940/ijrte.B1256.0782S319.
38. M. Batra and A. Bhatnagar, "Requirements Elicitation Technique Selection: A Five Factors Approach", *International Journal of Engineering and Advanced Technology (IJTEAT)*, vol. 8, issue 5C, pp. 1332-1341, 2019. ISSN: 2249–8958, DOI: 10.35940/ijteat.E1190.0585C19.

## AUTHORS PROFILE



**Krunal Bhavsar** is a research scholar and corresponding author for this research paper. He is pursuing his PhD in the stream of Computer Science and Engineering from Indus University (Ahmedabad, India). Krunal has received his MBA in Information Technology from Indus University (Ahmedabad, India) in 2018 and MSc in Mobile Application and Technology (Gold Medalist) from Gujarat University (Ahmedabad, India) in 2014 and BSc in Computer Science from Gujarat University (Ahmedabad, India) in 2012. He is an enthusiastic research scientist with 14+ years of corporate experience in software business process management, software project management, software engineering and data analytics. He is certified Scrum Master by Scrum.org for certifications 'Professional Scrum Master - I' and 'Professional Scrum with Kanban - I'. He is a Technical Project Manager in software research and development organization and he has professional expertise in data science and analysis also. He has been serving as a visiting lecturer at Computer Science department at Gujarat University also. Teaching, Training, Learning are his hobbies and he is passionate about research and development in Artificial Intelligence and Machine Learning technologies.



**Dr. Vrutik Shah** is a research guide and contributor author to this research paper. He has received his PhD from Karpagam University in the stream of Computer Science and MCA in the stream of Computer Application from Nirma University. He is a Technical Project Manager in multinational well-known software development organization – Streebo and he has been leading software process redesign and redevelopment with the help of trending automation technologies. He is a certified Agile Scrum Master and having expertise in software project and product management, business analysis and business process reengineering.





**Dr. Samir Gopalan** is a research co-guide and contributor author to this research paper. He is a director and a head of department of management, external relation manager and dean of faculty (OSD Research) at Indus University. He has achieved Ph.D., M.Phil. (Computer Science), MBA (Finance) and BSc degrees in his academic career. His domain research areas are marketing management, business

intelligence, training & development, skill & performance enhancement with 15+ years of experience.

<sup>1</sup>Krunal Bhavsar is the author to whom all the correspondence regarding this research paper should be addressed; E-mail: [krunalbhavsar@engineer.com](mailto:krunalbhavsar@engineer.com); [krunalbhavsar.rs@indusuni.ac.in](mailto:krunalbhavsar.rs@indusuni.ac.in). Tel: +91-9737007007