University of Sindh Journal of Information and Communication Technology (USJICT)
Volume 4, Issue 3, October 2020
ISSN-E: 2523-1235, ISSN-P: 2521-5582
Website: http://sujo.usindh.edu.pk/index.php/USJICT/

Towards medium-scale and large-scale game development using agile-based scrum

Danish Nazir Arain[1], Nazish Basir[2], Sehrish Nizamani[3]

[1]Institute of Information and Communication Technology, Faculty of Engineering and Technology, University of Sindh
[2]Department of Information Technology, Faculty of Engineering and Technology, University of Sindh, Jamshoro
[3]Department of Information Technology, University of Sindh, Mirpurkhas Campus
arain.danishnazir@gmail.com, nazish.basir@usindh.edu.pk, sehrish.basir@usindh.edu.pk

Abstract: Agile development is usually considered as a non-formal, transparent and fun-driven approach of software development, while scrum is an agile process within agile methodology. So often, the game development projects require huge teams and limited time schedules. Currently, the game industries are making a switch from the traditional approach of software development towards agile practices. This paper proposes a framework to be adopted and fitted in agile-based scrum framework for medium scale and large-scale game development projects. The proposed framework enlightens the scrum cast, the responsibilities and the overall process of scrum in detail. However, this framework is not tested and validated yet, but provides good insight to game development teams about how it can be applied.

Keywords: Agile; Scrum; Sprints; Game development; Medium scale; Large scale

I. INTRODUCTION

The game development industry is a huge and demanding industry. It is expected that the game industry which was reported with CAGR of $71.3 billion in 2015 is grown up to $90.1 billion in 2020 with a growth rate of 3.6 percent as a compound annual rate [1]. Beyond entertainment, this industry is also used for designing simulators in order to train soldiers, engineers, medical students and pilots [2, 3]. Designing a medium-scale or large-scale game is a complicated task that involves people with numerous expertise, with communication, coordination, and management.

The heart of Agile development is the transparency and interaction of the co-located teams [4]. Initially, Agile was designed for a small number of team settings but currently, this methodology is applied in large settings as well [5-7]. The game development industry is a complex industry that requires multi-disciplinary team members in order to complete the project [8]. Game development requires knowledge of mathematics, arts, artificial intelligence, modelling and graphics design [9]. These huge development projects with large teams when completed successfully return millions of dollars to the industry back.

Currently, for software development methodologies, either at large scale or small scale, there is more influence on adoptive developmental methodologies (lightweight such as Agile) which are rather than traditional predictive developmental methodologies (heavyweight such as waterfall) [10]. This fit of agile into a large-scale industry leads to some issues and challenges [7, 11].

The agile development in comparison with the waterfall model is based on implementation rather than documentation. Currently, agile is considered as a base from which numerous methodologies are derived such as Scrum, test-driven development, lean software development and feature-driven development [12]. The overall goal of agile development is to keep the project development in iteration and each iteration is individually treated as a mini project.

In this paper, an attempt is made to fit the Agile software development methodology into a medium or large-scale game development project. In doing so, a conceptual model is provided that lists out the possible agile team generation; and the responsibilities of each team member are outlined.

The remaining paper is arranged as follows. Section II discusses the related work, while Section III elaborates on the conceptual framework that fits the agile methodology into game development projects and also discusses the roles and responsibilities of the team members and the process of each scrum of game development. Section IV compares the proposed methodology with the practices adopted in game development studios. Section V discusses the learned lessons and possible limitations. Finally, Section VI present the conclusions and highlight the future work of this research.

II. RELATED WORK

There are numerous differences between traditional software development and game development in various phases of software development, such as software requirement, design, construction, testing maintenance, configuration management, engineering process, and engineering management [13]. Game development methodology was traditionally based on the waterfall model.

However, a recent study by Statcey and Nandkumar after interviewing 20 developers reported that these traditional software development models are harmful in the case of game development [14]. More game industries in Austria are shifted towards Agile practices [15]. Ruonala [16] made a systematic literature review of game development using agile methods and concluded that the game development industry is improved with the communication and collaboration of teams. However, the companies adopted and modified the agile methods according to their needs. Liu et al. concluded after a field study of 102 game development projects that the feedback of game tester regulates the outcome of sprint planning on the quality of the game [17].

Scrum-based development was initially proposed for small scale developments, but companies are now increasingly adopting agile methodologies for large scale developments as well. The agile development can be termed as large-scale if it involves more than two teams [18]. Few researchers modified the original scrum while including the middle layer as agents. Al-Azawi and his co-workers [10] proposed an approach of Agile game development that uses agent-oriented software engineering. This approach combines the features of both adaptive and predictive models of software development. Al-Azawi et al. [19] also proposed a multi-agent system engineering approach of scrum-based game development in which developers will be guided from the initial specification till system implementation. Various authors [20] [21] discuss the issues and challenges that occur in large scale scrum-based development projects. However, numerous challenges were reported, Evbota et al. [21] divide those issues into three areas such as (a) the ability of estimation, prioritization and planning, (b) the planning, working environment, and team management, (c) ceremonial agreement.

III. Proposed Scrum Framework for Game Development

An attempt to propose a framework to fit the Scrum into the large-scale game development industry. The framework includes the scrum cast and the scrum process that best fit in the medium/large scale game development industry. The proposed framework is depicted in Fig. 1.

The Figure 1 represents the proposed scrum cast and the modified process is presented in Section A and B, subsequently.

A. The Scrum Cast

The scrum cast includes the Product Owner (PO) and each category of the game development team.

For coordination and communication, the team should be small in a scrum [22]. This issue is resolved by dividing the members based on the features of the project in which they work. The scrum members in a scrum may include all members that work on that feature/level regardless of their expertise. Two approaches can be adapted while deciding the members of the scrum suggested below.

1) Level based design: Each team in a scrum works on a particular level of a game with the team of all required expertise level members.

2) Feature based design: Each team in a scrum works on a particular feature of a game with the team of all required feature expert members.

A Scrum Master (SM) is selected from each scrum and at no time he/she will be superior from the rest of the scrum team.

The PO in this large-scale scenario has the same role and is responsible for all cost, ROI (Return on Investment) related issues. He is responsible for providing the requirements for the product overall.

A new role is introduced here named as Project Coordinator (PC). The PC has the same responsibilities as SM, with the exception that the PC facilitates SMs with the information. He/She will communicate with all SMs in the duration of the project on which he/she is assigned to. The PC will also make sure there are no clashes between teams and that every task is being done on time.

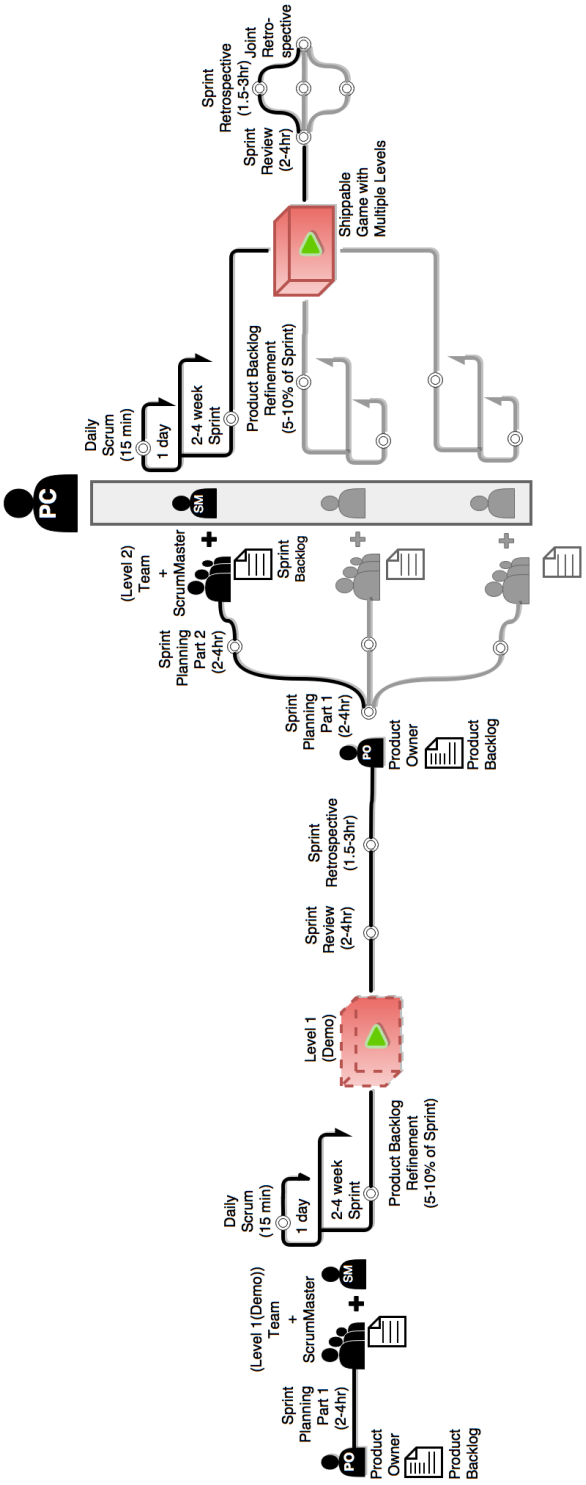

Figure 1. Proposed framework

B. *The Process of Scrum*

This section discusses the processes involved in the proposed framework of the scrum. The process starts with a single scrum followed by parallel scrums.

1) Product Backlog

It can be informally defined as anything given as a customer/PO such as a prioritized list of requirements for the game development. The PO is responsible for providing the ordering, contents, and availability of product backlog [22].

The product backlog should be designed with the suggestions provided by Mike Cohn [23], which he explained using the acronym DEEP, which refers to:

a) Detailed Appropriately: The items with higher priorities must be specified before lower priority items.

b) Estimated: Estimation of all items employing ideal days or story points should be provided. If the estimation is not possible for all items, only higher priority items should be estimated.

c) Emergent: The product backlog should allow amendments based on customer feedback.

d) Prioritised: All items should have priority values, based on which those items will be implemented first.

2) Daily Scrum Meeting

On each day of a sprint, a daily scrum meeting is held by a team. This meeting is daily held at the same location and same time. This meeting is usually termed as standup meeting and limited to almost 15 minutes of daily time. The members will discuss their game development work on a daily basis.

3) Sprint Planning Meetings

Once the product backlog is provided by the PO, it is reviewed in the sprint planning meetings. The goals and items on the backlog are discussed in order to have a full insight into the backlog.

Afterwards, depending upon the nature of the items in the backlog, the overall product backlog is divided into multiple area of sprints. For each area, a separate sprint backlog is designed that includes all the items related to it. The PC is in close communication with the PO and SMs in the duration of the sprint.

4) Sprint Backlog

Sprint backlog defines the sprint goal per team. The sprint backlogs are also supported in literature by [23] for large scale project developments. The sprint backlog is designed and managed by SMs and on the higher level by PC who is responsible for the avoiding clashes during the game project.

5) Sprint Review

At the end of each sprint, a meeting is organized, that aims to provide the demos of what is built during a sprint to the PO and other stakeholders. In this review meeting, all the sprint backlog items are discussed and at how much extent those items are covered. This meeting includes SMs, teams ' elective representatives, PC, and PO.

6) Sprint Retrospective

The sprint team along with the PC and SMs discusses the issues during the sprint period. Each member of the team suggests what is working well or not for the sprint in order to improve the planning of the next sprint. The PC collects the feedback of the team, for improving the upcoming sprit and communicating those issues in product retrospective.

7) Product Review

After each sprint meeting, a product meeting is organized. This meeting may include anyone interested including stakeholders, POs, customers, end-users, and interested users. This meeting focuses on providing the game demo what has been designed during all individual sprints.

8) Product Retrospective

All the SMs and PC gather and communicate the main issues occurred during their sprints and make plans to improve the upcoming sprint.

IV. Validation of proposed methodology with Scrum Practices

McKenzie et al. [24] made an exploratory survey in 2019 of 12 software engineering firms who adopted agile scrum for game development. The observed practices supported the small and medium-scale development.

The proposed methodology is compared against the practices reported in literature which are presented in Table I.

TABLE I. Comparision of observed practices with proposed methodology

Category	Response	Scrum Practices [24]	Proposed Methodology
Sprints	***Sprint***	90.9%	Yes
	Length (Weeks)	1.9 Weeks Average	2 - 4
	Sprint goals	100%	Yes
Sprint planning	***Planning***	72.27%	Yes
	Team commitment	100%	Yes
Standup Meetings	***When?***	81.18%	Daily
	Length (Minutes)	Mostly 1-15 mins but varies	1-15 mins
Sprint Review	***Review***	81.18%	Yes
Sprint Retrospective	***Retrospective***	70%	Yes
Scrum Roles			
Product Owner	***PO Responsibility***	100%	Yes
	Role defined	100%	Yes
Scrum Master	***SM Responsibility***	90%	Yes
	Role Defined	100%	Yes
Development Team	***Cross Functional***		
	Size	5- more than 12 average	Not defined but Sprint should run parallel with small teams
Scrum Artifacts			
Product Backlog	***Product Backlog?***	100%	Yes
	Ordered by	100% Most Value	Follows DEEP [23]
Sprint Backlog	***Sprint Backlog?***	100%	Yes
Product Increment	***Prototype cycles***	54.54% - yes 45.45% - No	yes

When compared with McKenzie survey given in Table I, we have confirmed that our proposed framework is a general framework that can be applied to small to medium scale game development studios.

V. Possible limitations and lessons to be learned

Due to the success and popularity of agile development projects, the companies are using these methods in medium and large-scale development as well. Medium and large-scale developments are in risk for late completions, cost overruns and failures of projects [25]. Due to this most companies use a mix of scrum along with traditional software development practices [17, 24]. Currently, as mentioned by after analysis of the postmortem notes of 20 game developments analyzes that gaming industry is currently using hybrid, iterative, ad-hoc and waterfall processes [26]. Even though the agile practices are now being adopted in gaming industries, but no actual framework is adopted, and the scrum is misjudged by most of the development teams.

It is important that medium and large-scale game development projects should follow a defined framework that is most effective in terms of its outcomes. However, in practice the agile scrum is applied with distinct practices with studio's own assessment and perfection of their practices [last]. As a limitation, there is a possibility that this framework will be modified and matured later as per need of the gaming studios.

VI. Conclusion and Future Work

In this paper, a conceptual framework of scrum-based game development is presented. Due to the iterative and interactive nature of scrum, it is best suited for game development-based projects. This framework is designed after analyzing the nature of the game development industry. It focuses on providing an initial working version of the game by using the standard scrum implementation. If the initial working version is approved by the PO and the rest of the stakeholders, the framework then suggests making parallel sprints of a scrum. This framework is not tested or validated in the real environment.

As scrum is designed for teams of few team members, fitting this framework in large scale development may lead to some open issues and research challenges., These issues are not discussed in the current study but are open for the research community in the field.

The other future work may include testing and validation of this framework in the real game design industry and to conclude whether this framework provides the advantages of adopting scrum at medium or large-scale game development.

References

[1] D. Takahashi, "PwC: Game industry to grow nearly 5% annually through 2020," Venture Beat, vol. 8, 2016.

[2] E. M. Raybourn, "A new paradigm for serious games: Transmedia learning for more effective training and education," Journal of computational science, vol. 5, no. 3, pp. 471-481, 2014.

[3] I. Dunwell et al., "Promoting healthy adolescent lifestyles through serious games: enacting a multidisciplinary approach," in Serious Games and Edutainment Applications: Springer, 2017, pp. 323-348.

[4] M. Fowler and J. Highsmith, "The agile manifesto," Software Development, vol. 9, no. 8, pp. 28-35, 2001.

[5] T. Dingsøyr and N. B. Moe, "Research challenges in large-scale agile software development," ACM SIGSOFT Software Engineering Notes, vol. 38, no. 5, pp. 38-39, 2013.

[6] J. Eckstein, Agile software development in the large: Diving into the deep. Addison-Wesley, 2013.

[7] T. Dybå and T. Dingsøyr, "Agile project management: From self-managing teams to large-scale development," in 2015 IEEE/ACM 37th IEEE International Conference on Software Engineering, 2015, vol. 2: IEEE, pp. 945-946.

[8] C. Yun, H. Panahi, and Z. Deng, "A Multidisciplinary, Multifaceted Approach to Improve the Computer Science Based Game Design Education: Methodology and Assessment," in Proceedings of the 47th ACM Technical Symposium on Computing Science Education, 2016, pp. 570-575.

[9] S. Foster and J. Brostoff, "Digital doppelgängers: Converging technologies and techniques in 3D world modeling, video game design and urban design," in Media Convergence Handbook-Vol. 2: Springer, 2016, pp. 175-189.

[10] R. Al-Azawi, A. Ayesh, and M. A. Obaidy, "Towards agent-based agile approach for game development methodology," in 2014 World Congress on Computer Applications and Information Systems (WCCAIS), 2014: IEEE, pp. 1-6.

[11] N. Sekitoleko, F. Evbota, E. Knauss, A. Sandberg, M. Chaudron, and H. H. Olsson, "Technical dependency challenges in large-scale agile software development," in International Conference on Agile Software Development, 2014: Springer, pp. 46-61.

[12] K. Pathak and A. Saha, "Review of agile software development methodologies," International Journal, vol. 3, no. 2, 2013.

[13] E. Murphy-Hill, T. Zimmermann, and N. Nagappan, "Cowboys, ankle sprains, and keepers of quality: How is video game development different from software development?," in Proceedings of the 36th International Conference on Software Engineering, 2014, pp. 1-11.

[14] P. Stacey and J. Nandhakumar, "A temporal perspective of the computer game development process," Information Systems Journal, vol. 19, no. 5, pp. 479-497, 2009.

[15] J. Musil, A. Schweda, D. Winkler, and S. Biffl, "A Survey on a State of the Practice in Video Game Development," Technical report, QSE-IFS-10/04, 2010.

[16] H.R. Ruonala, "Agile Game Development: A Systematic Literature Review," Computer Science, vol. 50, 2016.

[17] J.-W. Liu, C.-Y. Ho, J. Y. Chang, and J. C.-A. Tsai, "The role of Sprint planning and feedback in game development projects: Implications for game quality," Journal of Systems and Software, vol. 154, pp. 79-91, 2019.

[18] T. Dingsøyr, T. E. Fægri, and J. Itkonen, "What is large in large-scale? A taxonomy of scale for agile software development," in International Conference on Product-Focused Software Process Improvement, 2014: Springer, pp. 273-276.

[19] R. Al-Azawi, A. Ayesh, I. Kenny, and K. AL-Masrur, "Multi agent software engineering (mase) and agile methodology for game development," in 14th Middle Eastern Simulation and Modelling Multiconference, MESM, 2014, pp. 116-122.

[20] C. Larman and B. Vodde, "Scaling agile development," CrossTalk, vol. 9, pp. 8-12, 2013.

[21] F. Evbota, E. Knauss, and A. Sandberg, "Scaling up the planning game: Collaboration challenges in large-scale agile product development," in International Conference on Agile Software Development, 2016: Springer, Cham, pp. 28-38.

[22] H. Kniberg, "Scrum and XP from the Trenches," How we do Scrum, 2006.

[23] M. Cohn, Agile estimating and planning. Pearson Education, 2005.

[24] T. McKenzie, M. M. Trujillo, and S. Hoermann, "Software Engineering Practices and Methods in the Game Development Industry," in Extended Abstracts of the Annual Symposium on Computer-Human Interaction in Play Companion Extended Abstracts, 2019, pp. 181-193.

[25] S. K. Gupta, A. Gunasekaran, J. Antony, S. Gupta, S. Bag, and D. Roubaud, "Systematic literature review of project failures: Current trends and scope for future research," Computers & Industrial Engineering, vol. 127, pp. 274-285, 2019.

[26] C. Politowski, L. Fontoura, F. Petrillo, and Y.-G. Guéhéneuc, "Are the old days gone? A survey on actual software engineering processes in video game industry," in Proceedings of the 5th International Workshop on Games and Software Engineering, 2016, pp. 22-28.