Using the iStar Framework for Planning and Monitoring Sprints in Scrum Projects: An Experience Report

Renato Mesquita, Rafael Nascimento, Luana Souza, and Márcia Lucena

Departamento de Matemática e Informática Aplicada
Universidade Federal do Rio Grande do Norte
rmsnatal@gmail.com; rafael.jullian@gmail.com;
luana.tms@gmail.com; marciaj@dimap.ufrn.br

Abstract. Agile methods are known as an alternative to the traditional project management and for generally fragmenting the development process into several iterations of shorter cycles. In the Scrum method, for example, the term "sprint" is used to represent an iteration within the development cycle. In the Goal-Oriented Requirements Engineering (GORE), the requirements are described based on the actors' organizational goals, which are interpreted as the objectives to be achieved. In this context, some studies have been developed relating Scrum characteristics with GORE characteristics. Thus, this article proposes the use of the iStar Framework, a GORE approach, to aid in the planning and monitoring of sprints in Scrum projects.

Keywords: GORE, iStar, Agile, Scrum, Sprint Planning.

1   Introduction

Throughout the years, several project development methods have been presented and, among them, there are those categorized as agile methods [1]. These methods are known to be more adaptive and flexible than the traditional ones [2], and generally fragment the development process into several iterations of shorter cycles, like it is done, for example, in the Scrum method [3]. The Scrum, which is based on the principles behind the Agile Manifesto [4], uses the term sprint to represent an iteration with a fixed duration of up to four weeks within the development cycle. The purpose of a sprint is that, at its end, a version or the increment of a product is delivered to the client. As a result, Scrum establishes events (meetings) as a way to create a routine focused on the success of the sprints' goals [5].

The sprint begins with a planning meeting called the Sprint Planning Meeting and is finalized with the product review and the sprint retrospective meetings, which are known as Sprint Review Meeting and Sprint Retrospective. The Sprint Planning Meeting's goals are: (i) to prioritize the items to be developed and (ii) to define the tasks to be executed, thus composing the Sprint Backlog. On the other hand, the goals of the Sprint Review Meeting and the Sprint Retrospective are, respectively, to review the product and the process that was carried out [6]. Also, the Daily Scrum is conducted daily to follow up on the tasks performed and monitor the existence of obstacles that may hinder the achievement of the sprint's goals [5]. In Schwaber [7], it has been described that the Sprint Planning Meeting, the Sprint

Review Meeting, and the Daily Scrum are useful events for the examination of the progress of the sprints' goals. According to Deemer [8], because the sprints are time-boxed, there is enough flexibility for planned items to be removed during the iteration. However, the team must commit to delivering something ready (new increment), aligned with the planned goals.

Another context is the Goal-Oriented Requirements Engineering (GORE), in which the requirements are described considering the stakeholders' organizational goals, which are interpreted as the objectives to be achieved [9]. Some studies have been developed relating GORE concepts to the Scrum method. In Esfahani [10], the GORE was applied using the iStar Framework to portray social aspects of the Scrum process and thus help in identifying key factors for the success or failure in implementing the agile methodology in the organization. In Sen [11], a GORE-based elicitation technique for the extraction of the stakeholders' goals in Scrum approaches is proposed. In Scheideger [12], the integration between Scrum and the iStar was done aiming to map the organizational dependencies between the actors in the process. In Colomer [13], a set of iStar metrics for the management of requirements in agile methods was proposed. Finally, Araújo [14] offers the integration of the Tropos Framework with Scrum practices for goal refinement in user's stories. However, the use of the iStar Framework for modeling sprints in projects that use Scrum with the goal of understanding its possible benefits in the development process has not been explored in the literature.

Thus, due to the potential of the application of the iStar Framework in the modeling of features of the Scrum, this article proposes the use of the iStar Framework, a GORE approach [15] in its second version [16], to assist in the planning and monitoring of sprints in projects that use Scrum. For this, a report of its practical application in the development of a real system will be presented, highlighting the benefits identified in this proposal.

2 How can the iStar help in the planning and monitoring of sprints?

The team, seen as self-organized and cross-functional, is composed of the following roles: Product Owner (PO), Development Team (DT) and Scrum Master (SM). The PO is the person responsible for maintaining and prioritizing the Product Backlog; the DT is composed of professionals who provide an increment of the product at the end of each sprint through the execution of tasks; and the SM promotes and supports Scrum within the team [5].

Considering the elements that make up the iStar Framework, which are actors, goals, tasks, relationships, refinements and decompositions, and their models, the Strategic Dependency (SD) and the Strategic Rationale (SR) models [17], we can use them to help the Scrum team, for example, to plan sprints goals and serve as an artifact for the monitoring and management of the project. Thus, sprint goals can be refined into more specific goals for each member; and then be broken down into executable tasks, resources used, and dependencies between team members to meet the goals. This way, the application of the iStar model to each iteration of the process can provide several benefits to the team, as listed below:

B1. Assist the SM, PO, and DT in monitoring sprint activities, such as Sprint Planning Meeting, Sprint Review Meeting and Daily Scrum, for the observation of the sprint's evolution [7];

B2. The graphical representation of dependencies between team members can facilitate the reorganization of the sprint by the SM, since he/she will have a better vision for tasks prioritization;

B3. Understanding the individual goals of the DT actors through the refinement of goals, without losing focus on the sprint's goals established by the PO;

B4. Understanding the tasks that are executable by the DT, thus facilitating the definition of the Sprint Backlog. This helps to relate the tasks to the sprint goals established by the PO, which are aligned with the client's needs;

B5. Understanding the external actors and their relationships, thus helping to comprehend their influence and impact on the sprints; assisting, if necessary, in the reorganization of demands;

B6. Understanding the dependency relationships between agents that perform the role of DT, assisting in the reorganization and prioritization of team activities within the sprint.

The benefits listed for the use of the iStar models for modeling sprints in the Scrum method can affect the members in the Scrum team, depending on the role they perform (Table 1). Regarding B1, the iStar models can help the PO, DT, and SM to have a graphical view of the sprint and to monitor its progress. The visualization of the sprints allows the DTs to have a perception of the dependencies between goals and tasks (B3 and B4). Also, it can help find conflicts in these dependencies, thus facilitating the sprint reorganization by the SM (B2 and B6) in accordance with the DT and the PO. Finally, the graphical visualization helps to have an insight into how external agents, the sprint itself, and the development environment can positively or negatively impact the fulfillment of the sprint's tasks and goals.

Besides, models help in visual grouping, which allows the quick analysis of large amounts of information, assisting in the organization and presentation of data, and contextualizing details [19]. It also facilitates communication, understanding, problem detection, and visualization of possible errors and/or neglects, as well as allows exploring hypothetical scenarios and potential solutions [20] [21]. This way, iStar models can help solve challenges found in agile methodologies such as the negligence of performing validation without the client's perspective; control of changes in the development process and failure of communication between developers [22]. These challenges are also present in the daily life of professionals who use the Scrum [18] [24].

Table 1. Mapping of the proposal's benefits according to the roles of the Scrum team.

	B1	B2	B3	B4	B5	B6
PO	✓		✓	✓		
DT	✓	✓	✓	✓	✓	✓
SM	✓	✓		✓		✓

3 Applying this proposal to a real project

The proposal presented in this work was applied in the planning of sprints for the Legis Project, which was developed by the Court of Accounts of the State of Rio Grande do Norte (TCE/RN), a Brazilian public entity. The Legis Project consists of the development of a web system to automate the submission of municipal and state legislative norms of the institutions related to the TCE/RN, and for the registration of national legislative norms. Table 2 presents the Legis Project's requirements. Although the use of user stories is a more common practice in Scrum [18], use cases were applied because of the need for further formalization required by the institution.

Table 2. Legis system's use cases.

Code	Module	Description
UC_Legis_01	Legislation	Manage Norms
UC_Legis_02	Legislation	Search for Legislation
UC_Legis_03	Legislation	Register Norm with Validation
UC_Legis_04	Legislation	Validate Norm
UC_Legis_05	Administration	Manage Norm Categories
UC_Legis_06	Administration	Register Manual
UC_Legis_07	Administration	Register Resolution
UC_Legis_08	Public	Access Dashboard

Some sprint activities were captured from the project documentation and, mainly, from the Kanban board cards. This way, it was possible to model sprint activities to simulate development scenarios. Some sprints executed and their respective goals are listed in Table 3. Also, it is important to note that sprint goals do not necessarily represent features of the system, but any activity to achieve the purpose of the project. The "Application integrated with the institution architecture" goal associated with the S1 *sprint*, for example, was originated from the need to have a base application integrated with the institution's architecture so that the use cases presented in Table 2 could be developed from it.

The SD model (Figure 1) presents the Scrum team and its goal dependency relationships within the Scrum process. In this representation based on Esfahani [10], it is possible to observe the dependencies between the roles of the Scrum team actors, such as the presence of agents Lisa, Spike, and Dener, who belong to the DT.

Sprint Code	Goal Description
S1	Application integrated with the institution architecture
	UC_Legis_05 developed
S2	UC_Legis_01 developed
	UC_Legis_02 developed
	UC_Legis_03 developed
	UC_Legis_04 developed
S3	UC_Legis_06 developed
	UC_Legis_07 developed
	UC_Legis_08 developed

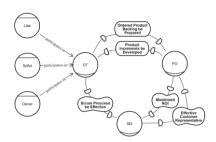**Table 3.** Sprint Backlog executed on the project and its goals.

Figure 1. SD model of the Scrum team, adapted from Esfahani [10].

Through the SD modeling of the Scrum team and the Sprint Backlog, we can make up the sprint's SD and SR modeling and use it in planning and monitoring. Figure 2 illustrates the SD model of sprint 1, characterized by the "Base application built and integrated with the institution architecture" and the "UC_Legis_05" goals. The goals can be observed by the dependency relations between the PO and the TD.
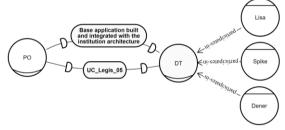

Figure 2. SD model of the Project Legis' sprint 1 goals.
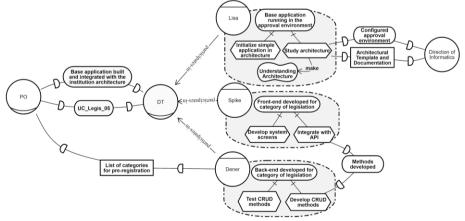

Figure 3. SR model of the Project Legis' sprint 1.

In the Sprint Planning Meeting, the PO and the DT are responsible for discussing sprint goals and predicting the tasks that will be developed [7]. At the end

of the meeting, it is possible to have an SR model representing how the team's activities were planned to achieve the goals set for the sprint (Figure 3).

It is possible to observe that the actors Lisa, Spike, and Dener have their tasks defined, represented by the decomposition of the goals of sprint 1. This model can help the DT to have an overview of the goals that must be accomplished in the sprint and the dependencies of tasks, goals and/or resources between the actors, helping to achieve better collaboration, communication, organization, and prioritization of activities.

4 Conclusions and future works

Research has been carried out relating the GORE approach with agile methods, in particular, the Scrum. In this sense, the iStar Framework, a GORE approach, was applied to assist Scrum teams in planning and monitoring sprints. Thus, the use of SD and SR models can bring several benefits related to the team's ability to improve its communication, organization, and the prioritization of its activities, helping stakeholders to have a graphical view of the team's goals and how to achieve them. Also, these models can be used to assist in monitoring the evolution of the sprints.

In order to continue this research, some suggestions for future work may be cited, such as: (1) explore the scalability of the iStar Framework for modeling sprint activities; (2) carry out a case study to obtain assessments on the advantages and disadvantages of the use of iStar models by the PO, the DT and the SM, as well as validate the benefits mentioned in this research; (3) elaborate heuristics for mapping the Sprint Backlog in SR models; and (4) develop tooling support with these heuristics and integrate the proposal with project management tools used in Scrum projects. Items (3) and (4) are related to the possibility of automating the creation of an SR model for each sprint, according to the concept of tasks that was already used in Scrum projects, thus diminishing any resistance from the team, which is already inserted in an agile context, regarding the development of iStar models.

Acknowledgment

References

1. Ambler, Scott. Agile modeling: effective practices for extreme programming and the unified process. John Wiley & Sons, 2002.
2. Aitken, Ashley, and Vishnu Ilango. "A comparative analysis of traditional software engineering and agile software development." 2013 46th Hawaii International Conference on System Sciences. IEEE, 2013.
3. Maximini, Dominik, Maximini, and Rauscher. Scrum Culture. Springer International Publishing AG, part of Springer Nature, 2018.
4. Beck, Kent, et al. "Manifesto for agile software development." (2001): 2006.

5. Schwaber, Ken, and Jeff Sutherland. "The scrum guide-the definitive guide to scrum: The rules of the game." SCRUM. org, Jul-2013 (2013).

6. Anand, R. Vijay, and M. Dinakaran. "Popular agile methods in software development: Review and analysis." International Journal of Applied Engineering Research 11.5 (2016): 3433-3437.

7. Schwaber, Por Ken. "GUIA DO SCRUM." (2011).

8. Deemer, Pete, et al. "A lightweight guide to the theory and practice of scrum." Ver 2 (2012): 2012.

9. Van Lamsweerde, Axel. "Goal-oriented requirements engineering: A guided tour." Proceedings fifth ieee international symposium on requirements engineering. IEEE, 2001.

10. Esfahani, Hesam Chiniforooshan, Jordi Cabot, and Eric Yu. "Adopting agile methods: Can goal-oriented social modeling help?." 2010 Fourth International Conference on Research Challenges in Information Science (RCIS). IEEE, 2010.

11. Sen, A. M., and K. Hemachandran. "Elicitation of goals in requirements engineering using agile methods." *2010 IEEE 34th Annual Computer Software and Applications Conference Workshops*. IEEE, 2010.

12. Scheidegger, M. E. S. *Integrando Scrum e a Modelagem de Requisitos Orientada a Objetivos por meio do SCRUM i**. Diss. Dissertação de Mestrado. UFPE–CIN, 2011.

13. Colomer, Daniel, and Javier Franch Gutiérrez. "Stargro: building i* metrics for agile methodologies." *Proceedings of the Seventh International i* Workshop: co-located with the 26th International Conference on Advanced Information Systems Engineering (CAiSE 2014): Thessaloniki, Greece, June 16-17, 2014 (CEUR Workshop Proceedings; 1157)*. CEUR-WS. org, 2014.

14. de Araujo, Leonardo Berbare, and Fábio Levy Siqueira. "Using i* with Scrum: An Initial Proposal." *iStar*. 2016.

15. Franch, Xavier. "The i⋆ framework: The way ahead." 2012 Sixth International Conference on Research Challenges in Information Science (RCIS). IEEE, 2012.

16. Dalpiaz, Fabiano, Xavier Franch, and Jennifer Horkoff. "istar 2.0 language guide." arXiv preprint arXiv:1605.07767 (2016).

17. Yu, Eric. "Modelling strategic relationships for process reengineering." Social Modeling for Requirements Engineering 11 (2011): 2011.

18. Darwish, Nagy Ramadan, and Salwa Megahed. "Requirements engineering in scrum framework." *International Journal of Computer Applications* 149.8 (2016).

19. Beatty, J., Chen, A.: Visual Models for Software Requirements. Microsoft Press, Washington (2012).

20. Horkoff, J., Yu, E.: A Qualitative, Interactive Evaluation Procedure for Goal and Agent Oriented Models. In: CAiSE Forum. CEUR Workshop Proceedings (2009).

21. Jaqueira, Aline, Bernardo Gurgel, and Márcia Lucena. "Integrando Requisitos ágeis com Modelos i*." ER@BR. 2013.

22. Medeiros, Juliana, et al. "Quality of software requirements specification in agile projects: A cross-case analysis of six companies." *Journal of Systems and Software,* 171-194, 2018.

23. Mohan, Aneesha. "Agile Project Management Challenges: Analyzing and Exploring Agile Project Management Challenges from a Practitioner Perspective: A case study on HMS." (2018).

24. Ashraf, Sara, and Shabib Aftab. "Latest Transformations in Scrum: A State of the Art Review." *International Journal of Modern Education and Computer Science,* 2017.