Introducing Scrum into the Development of a News Portal

VILJAN MAHNIC
Faculty of Computer and Information Science
University of Ljubljana
Trzaska 25, SI-1000 Ljubljana
SLOVENIA
viljan.mahnic@fri.uni-lj.si

Abstract: - We describe the experience with introducing Scrum-based software development process to the web applications department of the largest Slovenian publishing house. Scrum was introduced within the scope of the project of rebuilding the news portal of their major daily newspaper. The paper describes preparatory activities before the start of the project, the application of Scrum practices during the project, and lessons learned. The use of Scrum proved to be beneficial due to tight schedule and loosely defined user requirements that often changed during the development. Experience has shown that the appropriate assignment of Scrum roles, staff training, and strict adherence to the concept of “done” contributed significantly to the success of the project. On the other hand, the project also revealed some deficiencies in Scrum implementation, primarily the lack of release planning and inappropriate changes in team composition, which should deserve more attention in future projects. We hope that the experience described in this paper will help those companies that plan to introduce Scrum into their development process.

Key-Words: - Scrum, agile methods, software development, software project planning

1 Introduction

In spite of the fact that Scrum [1, 2] is the most widespread agile method (according to the last State of Agile Survey [3] it is used by 66% of 6042 respondents) the evidence about the challenges that arise when using it in industrial environment remains scarce. Dybå and Dingsøyr’s systematic review [4] of empirical studies on agile software development found only one study dealing with Scrum. Consequently, one of the clear findings was that the coverage of the research area should be increased considering Scrum an example of an area where there is a large gap and should be given priority.

The aim of this paper is to contribute to the evidence of Scrum use in practice by providing experience of the largest Slovenian publishing house, which introduced Scrum within the scope of the project of rebuilding the news portal of their daily newspaper with largest circulation.

The remainder of the paper is organized as follows: Section 2 describes the project setting and reasons for using Scrum. In Sections 3 and 4, respectively, preparatory activities before the start of the project and the use of Scrum practices during the project are explained. Section 5 presents lessons learned and recommendations for similar projects, and Section 6 provides a conclusion.

2 Project setting

The aim of the project was twofold: to renew the web edition of the company’s daily newspaper with the largest circulation in Slovenia, and introduce Scrum as the development process to their web applications department. Since the company did not have previous experience in agile methods, it was agreed that the author would help them in preparations for Scrum implementation and provide advice during initial Sprints.

The project lasted 7 months (from May until the end of November 2011) and consisted of 9 Sprints. The number of people working on the project varied slightly from Sprint to Sprint between 6 and 8. By rebuilding the web site the publishing company wanted to establish a new technological platform, renovate the content of the newspaper’s web edition, standardize further development and maintenance procedures, and facilitate editorial teams’ work. Drupal content management system [5] was chosen as the base for implementation. Fresh new look and some advanced technical solutions were expected to increase the number of portal’s users and their activities.

Scrum was considered appropriate development method for several reasons:

- Vague and changing requirements: It was clear from the very beginning that the initial
requirements were incomplete, and that some of them would change and/or expand during the project. Scrum simplifies the accommodation of emergent requirements and provides users with frequent feedback how the product actually works so that timely adaptations to the project can be made.

- **Project’s nature:** Agile methods are appropriate for development of web applications since this kind of applications require prompt adaptation to changing market and business needs.
- **Short time to deliver:** Company’s management imposed very short deadlines for project completion. Scrum seemed appropriate because it does not spend time on detailed upfront design, but concentrates on early and continuous delivery of working software. Strict prioritization of user stories assures that the most important functions are implemented first; therefore, only less important functionality remains unfinished in the case of delay.

It was expected that in the case of successful implementation Scrum would be accepted as a standard methodology.

### 3 Preparatory Activities

In order to take the full advantage of Scrum strengths and reduce the risk that its introduction would fail recommendations from [6] were carefully considered before the start of the project. Special attention was devoted to the assignment of Scrum roles, the definition of the concept of “done”, the choice of the Sprint length, the acquisition of a project planning and management tool, and the choice of an appropriate method for user stories estimation. All participants were trained in order to get fully acquainted with Scrum, and its strengths and weaknesses.

#### 3.1 Appropriate assignment of Scrum Roles

Scrum requires three roles: the Product Owner, the Team, and the ScrumMaster. The Product Owner is responsible for representing the interests of everyone with a stake in the project and its resulting system. He maintains the Product Backlog, a prioritized list of user stories [7] with estimated times to turn them into completed product functionality. The Team is responsible for developing functionality. Teams are self-managing, self-organizing, and cross-functional, and they are collectively responsible for the success of each iteration and of the project as a whole. The ScrumMaster is responsible for managing the Scrum process so that it fits within an organization’s culture and still delivers the expected benefits, and for ensuring that everyone follows Scrum rules and practices.

The role of the Product Owner is crucial for the success of a Scrum project. He/she communicates the vision of what is to be developed and defines the criteria by which it will be judged. It is important that the Product Owner provides timely answers to questions on details of user stories, and makes quick evaluations of work being done. A nonresponsive Product Owner can cause unproductive work periods, which make iteration planning more difficult or even impossible.

Therefore, it was very important that the associate editor of the newspaper’s web edition accepted the role of the Project Owner, while the head of the web development department played the role of the ScrumMaster. During his absence this role was performed by his assistant. Other project members were developers, constituting the Scrum Team responsible for implementation of desired functionality.

#### 3.2 Definition of “done”

Scrum strictly enforces the concept of “done”, which requires that each Sprint provides a potentially shippable increment of product functionality. The code should be fully tested and resistant to user errors in order to be used in practice.

In order to comply with this requirement a precise definition of “done” was developed before the start of the project, which required that:

- each user story passed all acceptance tests,
- the code was written in accordance with coding standards,
- each component (i.e., class, procedure, file) was adequately commented,
- appropriate documentation was written,
- the code was peer reviewed by another member of the Team,
- the functionality of the user story was approved by the ScrumMaster,
- the user story was accepted by the Product Owner.

#### 3.3 Sprint length

Scrum recommends the Sprint length of 2 to 4 weeks. Short Sprints provide more flexibility since changes in requirements can be considered in
shorter intervals, but increase overhead by requiring more frequent Sprint planning, review and retrospective meetings. The publishing company decided the Sprint length to be 3 weeks.

Each Sprint started with the Sprint planning meeting on Thursday and ended with the Sprint review and the Sprint retrospective meetings on Tuesday of the third week of the Sprint. In between the Scrum Team had 12 working days to develop software. At the end of each Sprint the Product Owner evaluated all implemented stories strictly considering the concept of “done”. All stories that did not conform to user requirements were rejected.

3.4 Project planning and management tool
Since the development team was partly distributed it was not possible to use the typical “wall” with user stories written on note cards. Instead, the web-based tool Agilo for Scrum was acquired for maintaining the Product Backlog and adapted in order to enable recording of time spent on each task (in addition to the amount of work remaining, which is required by Scrum). It was planned to pay special attention to data collection since it was expected that the empirical data collected during the project execution would contribute to better understanding and further improvement of the company’s development process. It was additionally agreed that the author could use these data for research purposes in order to evaluate his model for measuring performance of Scrum-based software projects [8, 9].

3.5 Effort estimation
Considering recommendations from the literature [7, 10] planning poker [11, 12] was chosen for estimating effort required for implementation of each user story. It was decided to use predefined values of 0.5, 1, 2, 3, 5, 8, 13, and 20 story points. A story point was considered to represent a working day consisting of 6 hours of effective work.

For the purpose of planning the Team also estimated its initial velocity. Having no previous experience, the initial velocity was simply computed as a product of the number of working days in the Sprint and the weighted number of developers, which considered the experience of each developer.

3.6 Training
Full commitment of all participants is a prerequisite when a new method or development process is introduced. Since most project members were not familiar with Scrum, a short internal training was provided in order to acquaint them with the method and ensure that each of them was fully aware of his/her responsibilities during the project.

4 Project Execution

4.1 Team composition
The core of the development team was composed of three developers that worked on the project from the beginning until the end. Two of them were familiar with Drupal from before, the third one used it for the first time. They established good relationship and worked quite well during the first four Sprints.

In the fifth Sprint the team was extended by an external collaborator who was responsible for web page layout conceptualization and design. He was unwilling to adapt to Scrum rules (especially Daily Scrum meetings and strong collaboration), which hindered teamwork and slowed down project progress. However, he was indispensable due to specific nature of his tasks.

In the same Sprint, another programmer was assigned to the Team for part time in order to develop interface with the company’s unified subscriptions system. Being unfamiliar with previous work and not knowing the Drupal development platform he represented more a burden than an aid to the rest of the team. Consequently, the velocity decreased significantly, and after two Sprints he was replaced by an experienced developer whose contribution was much greater.

4.2 Typical Sprint
Each Sprint started with the Sprint planning meeting that consisted of two parts. During the first part the Product Owner presented the highest priority user stories he wanted to be implemented during the following Sprint. The Team members asked questions in order to clarify details and be able to estimate development effort. Then the stories were estimated using planning poker and allocated to the Sprint so that the sum of story points fitted within the capacity determined by the Team’s velocity estimate.

During the second part of the meeting the Team decomposed each story into more detailed tasks and assigned responsibility for each task. Each team member individually estimated how many hours it would take to accomplish each task he had accepted.

Daily Scrum Meetings were held regularly every day at 9:00. Each team member reported what he had done, what he planned to do, and what impedi-
ments were on his way. Additionally, each team member had to record the amount of work spent and the amount of work remaining for each task he was responsible for. When a user story was finished it had to be documented and tested according to rules established by the definition of “done”.

The Sprint review and Sprint retrospective meetings took place on the same day. At the Sprint review meeting the completed stories were demonstrated to the Product Owner who (for each story separately) decided whether to accept, conditionally accept, or reject it. A story was conditionally accepted if it required only minor corrections in order to be completely done. The actual velocity of the Team was determined by computing the sum of story points of all accepted and conditionally accepted stories.

The Sprint review meeting was followed by the Sprint retrospective meeting at which each Team member was asked about the practices that benefited the Team or worked against it during the last Sprint, and to identify new practices that should be started during the next Sprint. After discussing all proposals the Team decided which practices should be improved/implemented during the coming Sprint in order to help them work better together.

4.3 Velocity Tracking

In order to monitor performance the Team carefully tracked its velocity. Comparison between the planned and actual velocity is shown in Table 1.

<table>
<thead>
<tr>
<th>Sprint</th>
<th>Planned velocity [story points]</th>
<th>Actual velocity [story points]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>32.5</td>
<td>30.5</td>
</tr>
<tr>
<td>2</td>
<td>34.5</td>
<td>32.5</td>
</tr>
<tr>
<td>3</td>
<td>35</td>
<td>26</td>
</tr>
<tr>
<td>4</td>
<td>40</td>
<td>31.5</td>
</tr>
<tr>
<td>5</td>
<td>39.5</td>
<td>7.5</td>
</tr>
<tr>
<td>6</td>
<td>42</td>
<td>29</td>
</tr>
<tr>
<td>7</td>
<td>32</td>
<td>32</td>
</tr>
</tbody>
</table>

While the first Sprints ran quite smooth, the actual velocity diminished significantly in the fifth Sprint due to the aforementioned two new developers added to the Team. They were expected to increase the amount of work completed, but created disruption instead, which decreased the productivity of other team members. A substantial difference between the planned and actual velocity in Sprint 6 was a consequence of too optimistic velocity estimate. Instead of adapting the estimate to actual achievement in previous Sprints the team succumbed to the pressure of approaching deadline and promised to deliver more functionality than actually possible.

4.4 Release Plan and Completion Date

Scrum requires the Release Plan to be created at the beginning of the project by allocating all user stories known into Sprints according to estimated Team’s velocity. In such a way the number of Sprints and the completion date can be (approximately) determined. However, in our project the initial Product Backlog was incomplete and it was not possible to create a reliable Release Plan. New stories were constantly added by the Product Owner at the beginning of each subsequent Sprint. Although it was expected that the portal will be launched after seven Sprints, this deadline was more a desire than a reliable estimate.

Consequences of emergent requirements are clearly shown in the Release Burndown Chart in Fig. 1, which indicates that the Team was not able to reduce the amount of work remaining as quickly as the Product Owner added new stories to the Product Backlog. At the beginning of Sprint 7 it became evident that the desired contents of the first release should be reexamined in order to get an acceptable completion date. After removing less important functionality the first release of the new portal was successfully launched after nine Sprints.

4.5 Relationship between the Product Owner and the Team

It seems that at the beginning of the project the Product Owner was rather surprised by the amount of work required on his part. Consequently, the initial Product Backlog was incomplete and at first the Team felt him not enough responsive. However,
he quickly became accustomed to his role and the relationship between him and the development team was very good later on.

The Product Owner independently managed the Product Backlog, defined priority of user stories, and wrote acceptance tests. The acceptance tests proved to be very useful, enabling the Team to easily make out what the Product Owner really wanted. In case of ambiguity the Product Owner promptly provided detailed explanation. He participated in all planning and review meetings communicating the vision of what had to be developed and evaluating the work performed. Throughout the project the mutual understanding and trust between the Product Owner and the Team increased and the amount of common knowledge grew.

5 Lessons Learned and Recommendations for Similar Projects

The introduction of Scrum-based development process was an important experience for all project participants. Informal conversations after completion of the project revealed that they considered the project a great success, which could not had been achieved without using Scrum. In this section we summarize some recommendations for similar projects on the basis of lessons learned and mistakes that we made.

5.1 Preparatory Activities

Introduction of Scrum (as well as any other software development method) should be well planned in advance. It requires full management support, adequate training of all participants, appropriate assignment of Scrum roles, definition of the concept of “done”, definition of the Sprint length, and estimation of the Team’s velocity. The aforementioned activities should be comprised within the Sprint 0, which should also include the definition of the initial Product Backlog, the initial estimation of all user stories known, and the creation of the Release Plan.

5.2 The Release Plan

In order to obtain a viable Release Plan the initial Product Backlog should be as complete as possible. The plan should be promptly updated when new user stories are added by the Product Owner.

The Release Plan helps the Product Owner and the Team decide how much must be developed and provides an estimate of the approximate duration of a project. It serves as a guidepost toward which the project team can progress. Without the concept of a release, teams move endlessly from one iteration to the next.

5.3 Release Burndown Chart

The Release Burndown Chart is a helpful tool for monitoring project progress. It makes visible the correlation between the amount of work remaining and the progress of the Scrum Team in reducing this work. The trend line for work remaining indicates the most probable completion of work at a given point of time and clearly shows the impact of new stories emerging during the project as shown in Fig. 1. In such cases the Release Burndown Chart can be used to simulate the impact of removing functionality from the release to get a more acceptable completion date.

5.4 Team Composition

The Team working on the project should not change, but must remain as stable and homogeneous as possible. Adding new members in the middle of the project can cause disruptions and decrease Team’s velocity, thus making a late project even later [13]. All members must accept and obey Scrum rules. A special attention must be devoted to coordination of team members who work in different areas.

5.5 Velocity Estimation

Velocity estimates should base on the actual velocity achieved in previous Sprints and must not increase under the pressure of approaching deadlines. Too optimistic velocity estimates lead to wrong expectations and false sense of finishing on time instead of giving a realistic picture of the project progress. If new people are added to the Team the estimates must take into consideration possible loss of efficiency due to increased need for communication and more complex interrelationships.

5.6 Concept “done”

A common definition of “done” contributes significantly to the quality of the product. It should be established at the very beginning and followed strictly throughout the project. It represents the project’s quality statement for a user story ensuring that the story is fully developed and tested, that it
works without errors, and that no more work is left to be done. Interested reader can find examples of definition of “done” in articles written by experienced Scrum practitioners, e.g. [14, 15].

5.7 Strict Adherence to the Method
When Scrum is used for the first time it is recommended to strictly follow its rules and practices. The method can only be adapted or modified when the development team has enough experience and knows well its strengths and weaknesses. All stakeholders must stick to the method and must not succumb to the pressure of deadlines.

In our case there were some deviations from the method which proved to be wrong. E.g., the idea of having the ScrumMaster and his proxy seemed practical, but required a lot of coordination between them. Similarly, the omission of release planning caused problems with regard to completion date. Detailed analysis at the end of project showed that the Sprints that were planned carefully and systematically ran much smoother and provided better results than the Sprints that were planned cursorily due to time pressure.

5.8 The role of Product Owner
As already stated in subsections 3.1 and 4.5, the role of the Product Owner is crucial for the success of the project. It is the Product Owner’s responsibility to consider which activities will produce most business value. For this reason, it was important that that this role was assigned to the associate editor of the newspaper’s web edition. Additionally, the Product Owner must be knowledgeable enough about Scrum to be able to write, maintain, and prioritize the user stories. Otherwise, the ScrumMaster must help him/her prepare and maintain the Product Backlog.

6 Conclusion
Agile software development methods have recently emerged as a new and different way of developing software as compared to the traditional methodologies. In spite of being the most widespread agile method, there is not much evidence about Scrum implementations in practice. Our study represents an attempt to fill this gap by describing the introduction of Scrum into Slovenia’s largest publishing house.

By describing appropriate preparatory activities before the start of the project, the application of Scrum practices during the project, and lessons learned the paper provides guidelines that could serve those companies that plan to introduce Scrum into their development process.

References: