

CSM Practicing Certification Renewal Assessment

Name: Gabrielle Benefield
email: gbenefield@gmail.com
date: February 14, 2005

Scrum depends on the inspect and adapt mechanisms of process control to manage the complexity of projects. For inspection to work, everyone must know what is being made visible. To implement the Scrum process, such regulating mechanisms as defined roles, involvement versus commitment, time-boxes, and regular cycles are used.

1. Describe one project on which you have used Scrum over the last twelve months. Describe:

The project was to develop a buddy list for a web based chat application.

- Purpose - what business goal was the project intended to deliver?
Increased subscriptions to the service, reduced churn as users have more reason to return to the site to see their buddies and increased usage of the chat application features which are highlighted through the buddy list.
- Length - what was the duration of the project?
2 months
- Cost - what were the budgeted and actual costs?
(Approximately) Budgeted \$180,000 Actual, \$ 200,000
- Value - what were the projected benefits and actual (if measured) actual benefits?
I do not have the numbers available but the actual benefits were greater than projected.
- Size - how many people were on the project team(s), how were they organized into teams?
2-3 internal developers, 1-2 QA, 1 Product owner, 1 part time operations person, 1 UI, 2 external consultants. There was 1 team.
- Teams - were the teams cross-functional and self-organizing? Were the teams collocated in an open space? Were the teams physically separated within one location, or located in more than one physical location?
The teams were cross-functional and organized themselves extremely well. The entire development team was on one open floor with large open tables. The team was mainly on one table (for the developers) with the UI, product owner and QA only a few feet away at another table.
- Initiation - how was the project initiated? How was the team trained to use the Scrum process?
The project was initiated through marketing. The project was on the company Roadmap and approval happened very quickly so we went from concept to planning and build very quickly.

The team had all been trained in Scrum with an initial kickoff by Ken Schwaber and consulting help from Chris Sepulveda. New people had been taught the process by the original team members.
- Reporting - how did you report progress to management and the customers?
The product owner would send email updates weekly or more often as the project neared completion, daily stand-ups were available to check on progress and weekly demo's were

given to demonstrate new functionality.

- **Change** - what difficulties were surfaced by Scrum that had to be resolved? How were these resolved?
Technical difficulties and blocks were identified quickly with Scrum so the feature set could be adjusted to meet marketing dates. UI changes that didn't get to the developers quickly enough were released in a later iteration as it was preferable to release a less pretty version to gain business value than wait on the latest changes to the interface. The initial spike to gain an understanding of technical difficulty proved optimistic and the full build was a lot more complex than initially thought so the full feature set was not released as early as originally intended. The release management process was inadequate and had to be improved to allow for new iterations to launch.
- **Management** - what was the previous role of the ScrumMaster? Who took on the role of Product Owner? To what degree were they successful in fulfilling their roles?
The Scrum Master was originally a web development manager.
The Product Owner role was taken on by a project manager with good product management skills. They were both very successful in fulfilling their roles and this continued to improve as they got used to the process.
- **Engineering** - what software engineering practices or environment had to be changed?
Scrum and XP both showed the inadequacy of many of the engineering practices and environment which is one of the reasons they had been bought in. It's harder to ignore the blocks with Scrum where typically you go around them. The development, QA and release process were changed significantly to improve our release speed.
- **Stabilization** - for how long did the software have to be stabilized before it could be released? How did you structure this stabilization process?
The application did have performance issues due to an inadequate QA load testing environment and this took some patches to get the chat application to work for most browsers and operating systems. The code itself was quite robust.
- **Success** - to what degree was the project successful? To what degree was the Scrum process instrumental in the success of the project?
It was successful and without Scrum it may not have been. The code would have been very fragile and un-maintainable and the UI and code may not have come together so easily if we had developed it in a typical fashion.
- **Scrum Process** - to what degree was the Scrum process implemented "out of the box?" To what degree did you have to modify the Scrum process for this project? For each modification, how did you formulate the modification so that the basic inspect/adapt mechanisms continued to function? What parts of Scrum couldn't be implemented, or failed, and why?
Scrum was modified as we used XP methods as well. We took the main tenets of Scrum, daily stand-ups, close collaboration, open development environment, cross-functional teams, regular updates to management and constant communication. The product team had a monthly planning cycle; the development team also did weekly planning at a more granular level to define the tasks. Monthly iterations work for product planning but not so well for our development cycle which is why we merged Scrum with XP. This worked but there is a danger that people see iteration cycles as points in time when products can be released so management had to be reminded that there was a difference between having working code and having a release that would give members a complete enough product to give value.

2. How do you cause the accuracy of Product Backlog estimates to improve? To what degree does their accuracy matter?

We moved towards measuring the teams velocity based on its output from the week before to determine how much work we could get done. Estimates are better than they were however the very rapid development schedule makes it hard to give the team enough time to fully understand the products and to do development spikes to understand the technical challenges. The pressure to release new products undermines the estimation process to some degree, when the pressure is not so intense the estimates are a lot better. We have developed a much more realistic roadmap than ever before and management is happier with this as they know they will actually get some products rather than over-optimistic promises where very little is delivered.

3. How do you cause the accuracy of what a team commits to for a Sprint to what the team actually delivers?

Giving visibility to the sprint goals and product backlog then ensuring everyone is aware of the progress helps a lot. Having the team understand that they all need to cross the finish line together and multi-task to do this also greatly improves success.

4. What metrics do you use to track the development process? Which metrics have been changed, removed, or newly implemented as a result of using Scrum?

We track ROI based on reduced churn, new sign-ups and any additional benefits. As we are able to be more predictable our financial models have also improved.

5. What type of training, resources, or tools would best help you successfully employ Scrum in the future?

The way we implemented Scrum worked extremely well for us. Over a few months we were able to completely change the way we developed products and this method was very successful. The combination of an organizational kick-off with Ken and having a knowledgeable consultant gives you a major advantage in transitioning quickly and well. Keeping a Scrum champion to lead and motivate the teams is also key to doing this well.

6. (Optional) Scrum and Extreme Programming are sometimes used together. What must be considered when this is done?

Scrum for us was a lot easier to implement than Extreme Programming (XP). For very fast and skilled developers XP was adopted quickly but for people less skilled it can be a steep and demanding learning curve. Merging two processes when the team has had major change already can be hard to do and should not be done too quickly. Ensuring that you have a good development environment and if possible an Object Oriented architecture where everyone knows the language is also important. The names for team roles are also different between Scrum and XP so a domain language needs to be established early on so as not to cause confusion. There is a lot of value of doing both Scrum and XP as they address different needs within development and it would be better to have a strategy upfront to deal with possible incorporation of XP using the best of both methods. We liked the display boards with cards on them that XP uses to track our projects and found that the burndown chart in excel from scrum didn't work at all for us and was discarded early on.