Q&A from February 2014 Webinar
Certified ScrumMaster (CSM) Round Table

Donna Knapp & Jayne Groll, ITSM Academy

Q: Do you have to have a background in software development to be a scrum master, or run projects in agile

A: No. Neither Donna or I are software developers. When we went through CSM training, we had a lot of epiphanies. While Scrum principles were originally applied to software development, it is now widely recognized that the same concepts can be used for any complex project. We practice Scrum principles at ITSM Academy for our content development.

Q: How to apply the release concept in Service Management, as at the end of the day we are not releasing any Software rather we deliver service management??

A: A release does not have to be a piece of software – we develop and release processes, procedures, business initiatives, etc.

Q: How does DevOps fit with the IT environments that use primarily of of box solutions. we do almost no development

A: Whether your purchase or develop software, there are still changes that are being made and released to your production environment. All commercial software is "developed" or customized; websites and web services are in a constant state of change. So DevOps and Agile principles help to identify bottlenecks or constraints within the software supply chain – from developer/provider through customization to build-test-deploy-operate.

Q: For multinational companies with huge service desks, how to apply the same Scrum Concepts for such Service Desks, knowing that the Scrum team is 7+-2?

A: A scrum team's sprint is working on a project, perhaps one that would benefit the Service Desk. The Scrum team should not be more than 9 people. But not all work goes into a sprint. The service desk's daily operation would not go into a sprint - it is basically unplanned work. You could flip a model and have 80% of your work be non-sprint operations (incidents) and the other 20% being worked on a Agile/Scrum-based project. Part of what scrum is really about is planning what work is going to be done by a given team in a given amount of time. It's time boxed so you're establishing a time frame (typically 2-4 weeks) where you want to focus and get things done. Hopefully priorities won't change dramatically in that amount of time.

Q: How about top Urgent, top priority Issues that require instant intervention, they are coming out of the plan (Sprint plan) how can we deal with such situations?

A: Once the elements of a sprint are agreed, it should not change. The product owner will set the priorities, and only the product owner can change priorities, abandon the sprint, do the reboot, and start a new sprint. However, a sprint is not 100% of your time. Unplanned, urgent work may be reduced but will never be eliminated. When planning a sprint, maybe plan for 80% of the time to the sprint, and 20% of the time to unplanned work. Or the other way around. It all depends on y our velocity – or ability to absorb work.

Q: I was under the impression that WIP limits applied more to Kanban but I just heard the speaker reference WIP limits in terms of Scrum. Is my way of think WIP is tied more closely to Scrum incorrect?

A: Your concept is absolutely correct. A Kanban is a way of making work visible, a tool that can be used in conjunction with scrum to make that work visible, and to make sure that everyone understands what everyone else is doing.

Q: But how does that work for a service management team? We support multiple products, with multiple product owners, in addition to dealing with production incidents, etc.

A: You can have multiple sprints going on at the same time for different products. Again, you need to consider the makeup of each team, their velocity (ability to absorb work) and current level of reactive, unplanned work. The goal of Scrum is to identify priorities, move forward incrementally and get things done.

Q: So, someone who just decides at the last minute that they didn't think of it during the sprint plan, but they want to add something once they are doing the acceptance testing?

A: No. One of the advantages of practices such as Scrum is that it helps to avoid scope creep. Sealing the goals and work of the sprint allows work to get "done". Continually adding or taking away tasks just creates more work in progress. If you got all your work done in a given sprint, you could then go back to your backlog and work on your next highest priority task. Only the process owner can change priorities or what goes into a specific sprint.

Q: So as per Donna's answer, then that way Sprint will be called-off each time, as Priority 1 is always coming for Huge Service Desks , which means that Scrum is not valid option for Service desks .... correct?

A: Remember, most Service Desk work is unplanned and operational. So that work would not fit the Scrum model. However, continual improvement efforts such as re-engineering and releasing the Incident or Problem Management process would fit Scrum very well.

Q: Is scrum good for managing support tickets for a software already deployed?

A: No – that work is unplanned and operational.

Q: How many people use scrum outside of software dev, is that still uncommon?

A: While it's very rooted in software development, certainly we are seeing a lot more use for complex projects. The interest in DevOps has also spurred more interest in making service management processes more agile.

Q: I believe there are a lot of variations to the Scrum framework, as it's a framework, and not methodology, so i believe there are many ideas towards applying into ITIL framework, and there should be a bigger framework combine both ..... what do you think?

A: While there are perhaps some different interpretations about how to apply Scrum, there are not really variations on the principles described in the Scrum Guide. There is definitely a relationship with ITIL and other ITSM practices. The bigger framework that would focus on both would be … Devops.

Q: How is it that Agile and Scrums facilitates/ensures that the work actually gets completed within the scrum cycle? Is it because of single-minded focus? Better, more reasonable estimating? ???

A: Each sprint has a clear definition of "done" for the work that's been included. Each element must contribute to what is known as a "potentially shippable product". That could be a feature, some other self-contained aspect. Our instructor said that "done" is when you do not need to think about it anymore. By focusing on smaller chunks of work that is "done", the entire project gets built incrementally instead of how a lot of unfinished work in progress.

Q: Can a project be too small to use scrum?

A: No – a very small project can be "done" in one or two sprints.