

Eight Tips to Become the Scrum Master Your Team Needs

by Mike Cohn • 42 Comments

Eight Tips to Become the Scrum Master Your Team Needs

As a Scrum Master, do you ever feel like you're not doing all you can for your team? Are you doing the basics of the job, yet feel you could do more?

This is a very important question to ask. We each have blind spots that prevent us from seeing some of things we could do to better help our teams.

In my work with great Scrum Masters, and those who may not be great yet but are working to get there, I've identified eight things people often need to be reminded of in order to become the Scrum Masters that teams need.

First Be Quiet

Embrace the effectiveness of the all important pause, the enabling power of silence. When you are tempted to speak, waiting first for others in that awkward uncomfortable stillness that follows a complicated question or thought-provoking comment. You don't need to have all the answers.

And while you are mentally silencing yourself, remember to listen actively. Resist the temptation to begin forming a mental reply before a team member has finished speaking. Instead, while someone else is talking, give their thoughts your full attention. When they've finished, think before replying. There's nothing wrong with being quiet while you decide how to reply.

Believe me when I say that to become a fully autonomous, self-organizing team, your team needs your silence more than it needs your insightful comments.

Feed Your Inner Responsibility Junkie

Although a Scrum Master does not assume responsibility for the success of the project—that charge remains with the team—a Scrum Master does assume responsibility for the team's adoption of Scrum and practice of it.

I often compare the Scrum Master role to that of an orchestra conductor. Both must provide real-time guidance and leadership to a talented collection of individuals who come together to create something that no one of them could create alone.

Or as Boston Pops conductor Keith Lockhart says, “People assume that when you become a conductor you're into some sort of a Napoleonic thing—that you want to stand on that big box and wield your power. I'm not a power junkie, I'm a responsibility junkie.”

So my second tip for becoming the kind of Scrum Master your team needs is to feed your inner responsibility junkie. Your responsibility is to make team members the best they can be. And part of doing that is to become the best that you can be.

Master the Art of Checking In

Third, Scrum Masters need to be aware of what's happening with the team without crossing the line into micromanagement. To do this they need to be able to check in without making it feel like they're checking up.

While checking up and checking in may seem similar, they are actually quite different. Scrum Masters who check in don't just ask about progress; they offer real help in removing any obstacles or distractions that are impeding that progress. They avoid blaming individuals at all costs. And, perhaps most importantly, they ensure the team is given complete autonomy to solve whatever problem they've been given.

Checking up is very different. When you check up on someone, you are looking for conformance to a plan--did the programmer finish the changes last night as she said she would? Did the tester automate the tests as promised or just run them manually "one more time" again?

Checking up is about getting status-type information. That's fine. But checking in goes beyond that to include offering to help remove any impediments to progress.

To be the kind of Scrum Master your team needs, practice and perfect the art of checking in without checking up.

Showcase the Team

Scrum Masters look good when their teams look good. As such, they do whatever is necessary to help the team achieve its goal. And when the team achieves something extraordinary, great ScrumMasters make sure everyone in the organization takes note.

Good Scrum Masters also help their teams celebrate their successes. The celebration doesn't need to be massive. You don't need to have The Rolling Stones play at your retrospective. (But if you do, invite me, please.)

Food is always a good way to celebrate. I've brought bagels, homemade cookies (yes, I can cook), or everyone's personalized favorite morning drink. An afternoon off to go to the movies together is great. I once took a team bowling, which was a blast because the alley displayed the speed of each ball. We paid more attention to that than to the number of pins knocked down.

You helped the team achieve their success. And that's very important. But they achieved it. Make sure they--and others--recognize and respect that fact.

So that's my fourth tip: To be the Scrum Master your team needs, make the shift in your own head from "Look what I did!" to "Look what I helped the team do!"

Clear a Path

Fifth, try to end each day with all known impediments resolved. That doesn't mean you can necessarily clear every impediment by the end of the day--some impediments take some time and maneuvering to remove. But you should make some progress every single day.

Remember, to be the Scrum Master your team needs you to be, you don't need to do the work--you need to make the work easier for the team to accomplish.

Be the Hardliner Your Team Needs You to Be

This sixth tip might seem counterintuitive. But sometimes you've got to be a bit of a jerk. By that I mean, you need to be unbending when it comes to certain rules and uncompromising in your confidence in the team's ability to find a way forward.

Here are just a few examples:

Never let the team team get away with taking partial credit for a story.

Don't let the team honor only the agile principles they find the most enjoyable. For example, many teams incorrectly interpret the Agile Manifesto to say that no documentation is needed. And they relish in that while treating working software as something nice to have but not necessary at the end of each sprint.

Don't solve problems for the team they should solve themselves. Sometimes working together to solve a problem helps a team identify ways to make sure the problem doesn't recur.

Push the team to improve. Are too many bugs escaping into production? That might be a good time to suggest incorporating new engineering practices: test-driven development, pair programming, continuous delivery or so on.

Insist on timely participation in every daily scrum. Be vigilant about getting everyone's input in team discussions such as sprint retrospectives.

Hold the team to account...every sprint...every time. Your team might not always like it at first,

but soon they'll come to recognize the benefits of holding steadfast to agile principles.

Know Your Stuff -- And Learn What You Don't

The best Scrum Masters have the technical, market, or specific knowledge to help the team in pursuit of its goal.

That means knowing enough about key technical issues to understand the problem and be able to explain it to others in the organization when necessary.

It also includes having a broad understanding of the market and the opportunities and challenges surrounding the product.

And it means knowing how decisions are made in the organization, who makes them, which coalitions exist, and so on.

So my seventh tip is this: If you don't have all of these skills, work to acquire them. This doesn't mean you need to become a developer or marketer. But you need to know enough about those types of work that you can assist resolving problems that affect those areas of your team's effort.

Assert Influence without Taking Over

That brings us to the eighth and final tip: Successful Scrum Masters know how to influence others, both on the team and outside it.

Initially, team members may need to be influenced to give Scrum a fair trial or to behave more collaboratively; later a Scrum Master may sell a team on the idea of trying on the idea of trying new technical practices such as test-driven development or pair programming. A Scrum Master should know how to exert influence without resorting to a command-and-control "because I say so" style.

Most Scrum Masters will also be called upon to influence those outside the team. A traditional team may need to be convinced to provide a partial implementation of their functionality to the Scrum team, a QA director may need to be persuaded to dedicate full-time testers to the project,

or a vice president may need to be talked into trying Scrum at all.

To be the Scrum Master your self-organizing team needs you to be, you need to do more than buy pizza and get out of the way. Learn how to influence the team and those surrounding the team in subtle and indirect ways. When you do, you can steadily move teams (and the organization itself) toward becoming more agile--which ultimately is what everyone really needs.

What Do You Think?

Whether you're a Scrum Master or team members, what reminders would you offer Scrum Masters who want to become the Scrum Masters their teams need? Please add your thoughts in the comments section.

Posted: May 30, 2017

Tagged: scrummaster, teams, daily scrum, scrum, influence
