

24 Common Scrum Pitfalls Summarized

by Mishkin Berteig

Scrum is the most popular agile method... if you count all of the teams doing “Scrum Butt”. Doing Scrum really well is much harder and much rarer. Here is a list of 24 common pitfalls or bad behaviours of Scrum teams:

1. **Excessive Preparation/Planning:** Regular big up-front planning is not necessary with Scrum. Instead, a team can just get started and use constant feedback in the Sprint Review to adjust it's plans. Even the Product Backlog can be created after the first Sprint has started.
2. **Focus On Tools:** Many organizations try to find an electronic tool to help them manage the Scrum Process... before they even know how to do Scrum well! Use manual and paper-based tracking for early Scrum use since it is easiest to get started. Finding a tool is usually just an obstacle to getting started. (And besides, check out what the Agile Manifesto says about tools.)
3. **Problem-Solving in the Daily Scrum:** The Daily Scrum should not be used to find solutions to problems (obstacles, impediments) raised. Instead, keep the meeting very short and have those problem-solving conversations afterwards with only those who are interested. The ScrumMaster facilitates this meeting to keep it on track.
4. **Assigning Tasks:** Even though the concept of self-organizing teams has been around for a long time, still some people think that a project manager or team lead should assign tasks to team members. It is better to wait for someone to step up than to “take over” and assign a task.
5. **Failed Sprint Restart:** Although cancelling a Sprint is rare, it can be tempting to try and wait until everything is “perfect” or “ready” before re-starting. Teams should immediately re-start after cancelling a Sprint.
6. **ScrumMaster As Contributor:** The ScrumMaster is like a fire-fighter: it's okay for them to be idle – just watching the team – waiting for an emergency obstacle. Taking on tasks tends to distract the ScrumMaster from the job of helping the team follow the rules of Scrum, from the job of vigorously removing obstacles, and from the job of protecting the team from interruptions.
7. **Product Owner Doesn't Show:** The Product Owner is a full member of the team and should be present at all Scrum meetings (Planning, Review and Daily Scrums). As well, the Product Owner should also be available during work time. Of course, the PO also needs to work with stakeholders and might be away during that time, but these discussions should be scheduled outside of the team's meeting times.
8. **Stretch Goals:** The team decides on how much work it will do in a Sprint. No one should bring pressure on the team to over-commit. This simply builds resentment, distrust and encourages low-quality work. That said, of course teams can be inspired by challenging overall project or product goals.
9. **Individual Heroics:** Individuals on a Scrum team should not do excessive individual overtime, or in any other way try to be the “hero” of the team. Scrum helps us build great teams of people, not teams of great people (quote from Barry Turner).
10. **Team Organizes Product Backlog:** The team does not have proper insight into the needs of users and instead should be focused on solving technical problems. The Product Owner needs to be accountable for ROI and therefore should resist any pressure from the team to do things in a particular order for “technical” reasons.
11. **Product Owner Specifies Solutions:** The Product Owner must allow the team full freedom to come up with solutions to the problems presented in the Product Backlog. PBIs should be free of technical specifications unless they can be tied directly to a customer or end-user request.
12. **Urgent Interruptions:** Urgent interruptions should not be allowed in a Sprint... instead, if it is

urgent enough, the team should cancel the Sprint. Otherwise, the interruption should be put on the Product Backlog and deferred until the start of the next Sprint.

13. **Making Assumptions:** Often team members will fail to ask the Product Owner about details of the work they are doing. A team member solves problems, but it is critical to know about constraints. The feedback between PO and Team Member should be ongoing throughout every day of the Sprint.
14. **Not Getting Done:** This is hard to prevent from happening from time to time, but it can become a habit for a team to over-commit. Make sure that a team that is doing this is using burndown charts effectively and is holding demos even when they have not completed all their work.
15. **Demo Not Ready:** Sometimes a team forgets that it will take time to prepare for a demo: cleaning the team space, setting up a demonstration environment, getting scripts ready, and ensuring that critical stakeholders are prepped. These activities can be part of the tasks in the Sprint Backlog.
16. **Prototype Not Shippable:** A Scrum team should attempt to produce production-quality, “potentially shippable” software right from the very first Sprint. Building prototype code just delays the inevitable need to write production code. Similarly, wireframes, detailed designs and other similar tools should also be avoided.
17. **Distributed Team:** Although Scrum does not officially require team members to be collocated in a “war room”, the reality is that any distribution of team members (even just into separate cubicles) has a huge negative impact on transparency and communication, which in turn has a huge impact on productivity and quality.
18. **Directive ScrumMaster:** The ScrumMaster needs to be a facilitator who supports the team in learning self-organization, and the Scrum rules. The ScrumMaster should never succumb to the temptation to suggest how a team member does his or her work, nor what task next to work on.
19. **Changing Team Membership:** Scrum is a framework for creating high-performance product development teams. If the membership of a Scrum team changes, it forces that team to re-start the Forming-Storming-Norming-Performing sequence. If the team is in Norming or Performing, then changing team membership for any reason is a waste of quite an investment.
20. **Non-Scrum Roles on the Scrum Team:** It is very common for an organization to create a Scrum team without changing the official title and duties of the people who are members of that team. For example, a person who is a Project Manager might be given the responsibilities of the Product Owner without an official change of title. Scrum teams should only have a ScrumMaster, a Product Owner, and Team Members.
21. **Giving Up On Quality:** Scrum has very high demands on teams regarding the quality of their results: “potentially shippable software”. It is easy for a team or organization to fall back on the crutch of defect tracking software instead of maintaining extremely high levels of quality at all times (due to pressure to release features).
22. **Imposed Deadline Scope And Resources:** Scrum is reality-based. If an external stakeholder wants to impose a minimum scope, a deadline and constrain the resources available to the team then they must allow that quality will slip... which in turn is against the principles of Scrum. The reality is that no-one can predict the future so any such imposition is simply a fantasy.
23. **Definition of “Done” Imposed:** There is confusion between the concept of the definition of “done” and “standards”. Managers and other stakeholders often incorrectly impose standards on a team as its definition of “done”.
24. **ScrumMaster Not Present:** I once saw an organization that had created a room for a “team” of ScrumMasters. They worked there together most of the time! The only time the ScrumMaster should be away from the team is when he/she is working on removing an impediment that is outside the team. Otherwise, the ScrumMaster should be in the team’s room.