PERFORCE

7 ANTI-PATTERNS FOR SELECTING AN AGILE TOOL

FOR PRODUCT DEVELOPMENT

Introduction

Are you considering a new Agile project/product management tool for the team? There's a lot at stake.

Managers will be basing critical decisions on its data. Team members will be using it every day to understand what they should be working on. They'll also be using it to understand how they fit into the organization. In other words, to understand that their work matters.

Today, many Agile tools claim to facilitate agility, enhance collaboration, and ship better products with less stress. When selecting one from that crowded list, how do you make the best possible choice?

A better question: how do you *not* make a *bad* choice?

To help you (and drawing from experience with numerous product owners, project managers, and teams), here is a list of often-made, easily avoided mistakes. Some of which *may seem like they are okay*, but are, in fact, not. We call them "anti-patterns."

An anti-pattern is a common response to a problem. It's usually ineffective and often counterproductive.

4 Goals of an Agile Tool

When picking a tool for product development, you need a solution that:

1. Is easy to use (and for a wide variety of users).

Pick something that's neither too complex for the novice or non-technical user nor too lightweight to get the job done. Ideally, your next tool will make life easier for everyone.

2. Scales without performance issues.

Your new solution should be fast and efficient if there are to be hundreds, or even thousands, of real-time users. In such cases, a few extra clicks or seconds of loading time add up to lost hours, even days, when scaled across an organization.

3. Supports task management and big-picture planning.

Your next tool should accommodate the team, product owner, and other high-level stakeholders. It should bring together the many vital activities of product development while supporting organizational alignment, collaboration, and innovation.

4. Allows you to work your way – not how textbooks say you should.

Don't be boxed in by any single method pushed by a particular tool, blog, or textbook. You may want to switch methods in the future, or even explore others. In which case, you won't want to suffer through learning another delivery method and the hassle of another re-tooling.

Finally, keep this in mind: Agile, at its core, is about minimizing tooling. So, if you're at the beginning or in the midst of implementing Agile methodologies, take care not to pick a tool that impedes your Agile iteration and delivery – either now or in the future.

7 Anti-Patterns for Selecting an Agile Tool

⊗ Pick something popular.

From cola beverages to Agile tools, certain products become slightly more widespread. Some because of branding or marketing. Others ride the wave of familiarity. Atlassian's Jira, which is technically an issue tracking software, and Trello, a board tool for small teams, are two such tools. Many teams may begin adopting them simply because they dominate the shortlist of tools they are aware of.

If you use such tools without doing a more extensive analysis and considering potential drawbacks, you may be limiting your potential. For example, can the popular tool handle complexity with the nuance you need? Instead, start with a full field of options that includes popular, best-of-breed, and niche tools.

⊗ Pick the first tool suggested.

Sometimes we have one passionate team member who either begins (or even leads) the Agile tool conversation. Now, letting the team provide input on tools they'll use is a great practice. Make sure, though, that the team has seen more than one tool before you make the final choice. There are many users and stakeholders to consider. To pick the first tool that enters the dialogue is a mistake. Instead, solicit input from the team and, naturally, include your own thoughts.

⊗ Don't include the team.

A decision-maker may, to speed up implementation, gather minimal input from the team. Perhaps they had a good experience with a tool previously and want to keep using it. Or maybe they've conducted extensive preliminary research and feel confident (and justified) in their decision.

Even so, remember that every single team, circumstance, and organization is different. While a given Agile tool worked there and then, it may not work here and now. A short pilot or trial phase is common, and is excellent for either unearthing issues you didn't expect, or for confirming what you expected.

⊗ Select a tool for the team you currently have.

What might your team or environment look like in six months? Two years? Five? 10? An Agile tool is a long-term investment. Consider your concrete near-, mid-, and long-term needs when evaluating your solution. Ideally, it will scale with your best, most wildly successful scenario. Think: how will tomorrow's team succeed (or fail) using the tool you select?

⊗ Let the tool guide your agility.

Whatever solution you choose will shape your team. But don't make the mistake of locking them into a process dictated by a tool.

Does your team actually work better using sprints? Or is it just because that's what they're used to? What if one project could be better using another delivery method? What can Kanban boards do to improve your scrum team?

You wouldn't know of the potential for productivity in these cases because an alternative was never presented. Keep in mind that your tool will shape how your team works together in many ways. The best Agile tools facilitate the most natural processes for your team, product, and contexts. The best Agile tools stay out of your way.

⊗ Compare monthly license fees only.

Make no mistake, comparing cost is critical to your evaluation. It's when monthly user costs are calculated alone that we run into trouble. When we take cost (a monthly licensing fee, for example) at face value, we can miss hidden costs.

For example, one of the popular tools today, Atlassian's Jira, seems at first glance to be an inexpensive tool. To make Jira function as an enterprise Agile planning tool, however, you must add several plugins. This solution can be costly to scale. Instead, detail all costs, including any upgrades or plugins you would need to achieve the functionality you want.

Finally, what (and how many) software prerequisites does your Agile tool have? Whether databases, operating systems, or browsers, these prerequisites can increase the total costs of your solution as well as delay installation and upgrades.

⊗ Sneak your choice past IT.

In the end, it may not matter how excited you are about a particular Agile tool if your IT team doesn't approve. If IT needs to sign off (and even if it doesn't), you'll want to make sure your tool doesn't expose intellectual property. Because if you're building a successful product, you should have some well-founded expectations for on-premises security.

So bring IT on board early. They may put up resistance to certain tools, but it's often just because they're viewing (and judging) your solution from another angle. IT also plays a significant role in helping you determine the total costs mentioned in item six above.

7 Better Practices

Avoid the anti-patterns! If you're looking for a viable, long-term Agile solution, consider these best practices instead:

- ✓ Add best-in-breed tools and commodity tools to the list.
- ✓ Solicit opinions from a wide range of stakeholders.
- ✓ Let team members evaluate and battle test tools back-to-back.
- ✓ Compare tools with your short- and long-term goals in mind.
- ✓ Pick a tool that flexes to your needs and context (not the other way around).
- ✓ Calculate total cost – not monthly fee alone.
- ✓ Engage your IT department for a thorough vetting.

User Checklist

Many different people will be using the Agile tool every day. What really matters to them? To help you match needs across different roles in your company, here is a checklist organized by user role and written as user stories. Consider it a starting point for discussions with stakeholders.

As a Product Owner…

- ■ I have a single dedicated product backlog for backlog management and team transparency.
- ■ I can structure the backlog to communicate product vision and goals.
- ■ I can prioritize items using the best method (e.g., MoSCoW, WSJF) for my context.
- ■ I can gather feedback from stakeholders and easily maintain their wish lists.
- ■ I can easily create reports and dashboards for a big-picture understanding.

As a Project Manager…

- ■ I can schedule using one or more mechanisms (Gantt, Scrum, Kanban) to optimize delivery.
- ■ I can see real-time updates from everyone to understand true status.
- ■ I can quickly create dashboards and reports for the things that matter to me.
- ■ I have visibility into multiple projects to conduct a cross-project analysis.
- ■ I can develop standardized workflows and pipelines, so the team follows agreed processes.
- ■ I have estimation, time tracking, and other tracking features to understand costing.
- ■ I can manage dependencies among work items to understand potential issues.

As a Team Member or Contributor…

- ■ I clearly understand my priorities across projects to complete what matters most.
- ■ I can update status and work remaining quickly and easily.
- ■ I can participate in planning through collaborative features.
- ■ I can integrate with my version control for a streamlined workflow.
- ■ I can collaborate with other users and minimize use of other tools (e.g., email, chat).

As an IT Professional…

- ■ I can securely deploy the solution (e.g., on-premises) while complying with the security requirements I need.
- ■ I can configure access security controls to protect intellectual property.
- ■ I have 24/7 access to customer support resources to troubleshoot any issues.

Make Teamwork Click

Teamwork clicks into place when you have the right Agile tool. Team members know what they should be doing. They feel connected to each other and the organization they're contributing to. Product owners and managers can collect and manage input and transform it into a plan – a plan they can track, measure, and execute. (Then iterate, iterate, iterate!)

The end result? Valuable products get shipped to happy customers. Hopefully, by providing these anti-patterns and insights, this document will help you find the best Agile tool for your organization, and steer your team toward delivering exactly that.

Consider adding Hansoft to your list.

TRY HANSOFT TODAY FOR FREE

Hansoft