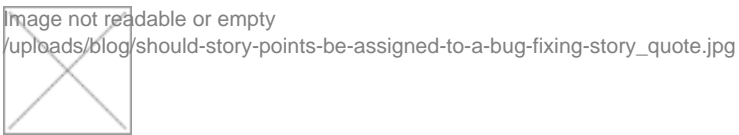


Should Story Points Be Assigned to a Bug Fixing Story?

by Mike Cohn •

78 Comments



Sometimes teams write a user story for this activity such as: “As a user, I want at least 15 bugs fixed” or, “As a user, I want you to spend about 50 hours this sprint fixing bugs so that the application becomes gradually more high quality.” Even a team that doesn't explicitly write such a user story will usually include a row on its taskboard to make the agile defects and bug fixing visible, and to track it.

In these situations, a common question is whether the team should assign some number of story points to the work of fixing these legacy bugs. If the team does not assign a story point value to this work, velocity will show the amount of “forward progress” the team is making each sprint. This has the advantage of making visible that the team is going more slowly through the work than it could if these legacy bugs had been fixed when originally found. On the other hand, if the team does assign points to the agile defect bug-fixing effort, velocity comes to represent the team's true capacity to accomplish work.

When migrating a product from a traditional approach to an agile one, teams commonly bring along a large database of agile defects. These are the result of an inadequate focus on continuous high quality. Shortly into their agile initiative, many teams (and their product owners) decide to aggressively work through the bug backlog. A common approach for doing so is to plan to fix X bugs per sprint or spend Y hours on bugs per sprint.

My usual recommendation is to assign points to bug fixing the agile defects. This really achieves the best of both worlds. We are able to see how much work the team is really able to accomplish, but also able to look at the historical data and see how much went into the bug-fixing story each sprint.

Knowing this can be helpful to a team and its product owner. For example, imagine a situation in

which the product owner is considering suspending bug fixing for the next six sprints in order to add a valuable different feature into a release. If we know that the team's full historical average velocity was 25 but that 5 points went to bug fixing each sprint, we know that suspending bug fixing for the next six sprints will result in 30 (6*5) more points of new functionality.

Posted: November 13, 2010

Tagged: product owner, sprinting, estimating, story points, defect management, continuous improvement, taskboard

About the Author

Mike Cohn specializes in helping companies adopt and improve their use of agile processes and techniques to build extremely high-performance teams. He is the author of *User Stories Applied for Agile Software Development*, *Agile Estimating and Planning*, and *Succeeding with Agile* as well as the [Better User Stories](#) video course. Mike is a founding member of the Agile Alliance and Scrum Alliance and can be reached at hello@mountaingoatsoftware.com. If you want to succeed with agile, you can also have Mike email you a short tip each week.
