

Velocity-Driven Sprint Planning

by Mike Cohn •

18 Comments

There are two general approaches to planning sprints: velocity-driven planning and commitment-driven planning. Over the next three posts, I will describe each approach, and make my case for the one I prefer.

Let's begin with velocity-driven sprint planning because it's the easiest to describe. Velocity-driven sprint planning is based on the premise that the amount of work a team will do in the coming sprint is roughly equal to what they've done in prior sprints. This is a very valid premise.

This does assume, of course, things such as a constant team size, the team is working on similar work from sprint to sprint, consistent sprint lengths, and so on. Each of these assumptions is generally valid and violations of the assumptions are easily identified—that is, when a sprint changes from 10 to nine working days as in the case of a holiday, the team knows this in advance.

The steps in velocity-driven sprint planning are as follows:

1. Determine the team's historical average velocity.
2. Select a number of product backlog items equal to that velocity.

Some teams stop there. Others include the additional step of:

3. Identify the tasks involved in the selected user stories and see if it feels like the right amount of work.

And some teams will go even further and:

4. Estimate the tasks and see if the sum of the work is in line with past sprints.

Let's look in more detail at each of these steps.

Step 1. Select the Team's Average Velocity

The first step in velocity-driven sprint planning is determining the team's average velocity. Some agile teams prefer to look only at their most recent velocity. Their logic is that the most recent sprint is the single best indicator of how much will be accomplished in the next sprint. While this is certainly valid, I prefer to look back over a longer period if the team has the data.

I prefer to take the average of the eight most recent sprints. I've found that to be fairly predictive of the future. I certainly wouldn't take the average over the last 10 years. How fast a team was going during the Bush Administration is irrelevant to that team's velocity today. But, similarly, if you have the data, I don't think you should look back only one sprint.

Step 2: Select Product Backlog Items

Armed with an average velocity, team members select the top items from the product backlog. They grab items up to or equal to the average velocity.

At this point, velocity-driven planning in the strictest sense is finished. Rather than taking an hour or two per week of the sprint, sprint planning is done in a matter of seconds. As quickly as the team can add up to their average velocity, they're done.

Step 3: Identify Tasks (Optional)

Most teams who favor velocity-driven sprint planning realize that planning a sprint in a few seconds is probably insufficient. And so many will add in this third step of identifying the tasks to be performed.

To do this, team members discuss each of the product backlog items that have been selected for the sprint and attempt to identify the key steps necessary in delivering each product backlog item. Teams can't possibly think of everything so they instead make a good faith effort to think of all they can.

After having identified most of the tasks that will ultimately be necessary, team members look at the selected product backlog item and the tasks for each, and decide if the sprint seems full. They may conclude that the sprint is overfilled or insufficiently filled and add or drop product

backlog items. At this point, some teams are done and they conclude sprint planning with a set of selected product backlog items and a list of the tasks to deliver them. Some teams proceed though to a final step:

Step 4: Estimate the Tasks (Optional)

With a nice list of tasks in front of them, some teams decide to go ahead and estimate those tasks in hours, as an aid in helping them decide if they've selected the right amount of work.

In next week's post, I'll [describe an alternative approach, commitment-driven sprint planning](#). And after that, I'll share some recommendations.

Posted: October 21, 2014

Tagged: product backlog, velocity, sprint planning, sprint backlog, tasks, estimates

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.
