Table of Contents

Why Do We Use Story Points Instead of Hours or Days?

Story Points should be used instead of absolute hours because people are naturally better at relative estimation. Relative estimation with story points:

- Take less time,
- Provide enough accuracy and precision for organizations and Teams to forecast and plan,
- Can scale from individual Teams to Team-of-Teams and Portfolios.

But Story Points are not equal to hours. A lot of people like to equate hours or days with Story Points. Sometimes this is because there:

- Is a misunderstanding of what Story Points are, or
- Is starting guidance we get when first doing estimates, or
- Are Teams that have small, well known, low complexity work where they actually know how many hours it takes them to implement.

If, for example, you are using SAFe, and have never done the Story Point approach before, SAFe introduces the idea of standardizing points where “the shared understanding of 1 story point 'is '~ 1 day of effort to develop & test a story'”. This ensures we were able to do reporting at higher levels by rolling up numbers. SAFe also establishes the initial Velocity for the Team by multiplying the number of people on the Team (not including PO and SM) by 8 (days) and then adjusting the result down by subtracting the number of days that people would not be available during a Iteration. The training material indicates that this was a starting point.

The problem is that many people are still using this understanding when estimating Stories by saying things like “a day is equals a point and so since this item is about 10 days, the estimate is 10 points”. They then fudge the number to an 8 or 13 because “we are only allowed to use the Fibonacci numbers.” This is not the intent, and if you head in this direction there is no benefit to using Story Points.

To be clear, if the estimate is really just a proxy for time (hours or days), we should stop calling them Story Points. After all, they are just “ideal days” or “ideal hours” with all the inherent problems. Daryl Kulak says:

> If a development team is equating some number of hours to a storypoint, it is missing the point (pun intended). Saying, “Thirty hours equals one point” is ridiculous. If you’re doing that, just use hours for goodness sake. You haven’t gained anything from storypoints except to be able to claim to be doing “agile.”

Here are some things to understand when using Story Points instead of a more traditional approach to estimating.

Story Points and Velocity

Team Velocity is expected to increase over time as the Team learns how to deliver value more effectively. If this is grounded in time-based estimates, you cannot expect Velocity to increase as the reality is that there is only so much time available in a two week period (80 hours, right?). Velocity numbers usually increase over time as the Team learns how to work together more effectively and so their effective capacity increases.

Story Points Are More Than Size

Sure, “how long” you expect something to take is part of the estimate, but you also want to factor in risk / uncertainty, and complexity. For example, if the work looks like it will take the Team a day to do, but there is a lot of risk involved, then the Team might want to give a higher (not 1) estimate – say a 2 or a 3. If the duration is expected to be 8 Team days, and it is low risk and complexity, the Team might consider the estimate to be an 8, or the Team might say “we are rubbish at estimating week long efforts so let’s give it a 13 to address this uncertainty.” In this example the 13 hopefully leads to a discussion about splitting the story to improve its chances of delivery.

Story Points Need a Baseline

Estimates should be relative to something; a known piece of work. Start with a 1 sized piece of work (about a day, low risk, low complexity, and low uncertainty) that the whole Team can understand. Then, as you do other estimates the Team talks in terms of “in comparison to this 1 sized piece of work, I think this is about 3 times bigger, so this is a 3.” Perhaps you also identify another sized piece of work, say a common understanding of an 8 sized story. Having an understanding of two different sized stories will help the Team triangulate the estimate for any new stories – “its bigger than this 1 sized story, but smaller than this 8 sized story so it a …” Some Teams call these known stories their “keystone” stories.

Estimation is Unique to Each Team

Each Team will have a different view of the factors that contribute to “medium risk” but within a Team they will have a shared understanding. So one Team might say “if it involves new screen then, at a minimum, it is an 8 because we need to work UX issues, work closer with the customer, …” Another team might have UX expertise and so this may not be a reason to differentiate for that team. But they may have an approval process that seems to slow things down, so stories that have this approval step are larger. Each team will have a different view of the factors that effect “size” of work for them.

Estimation is Continually Refined

Over time, as Teams do more estimating, the Team will start to build up a table in their heads:

Duration	Risk	Complexity	Uncertainty		Story Points
Low	Low	Low	Low	→	1
Low	Low	Medium	Low	→	1
Low	Medium	Low	Low	→	2
.	.	.	.	.	.
.	.	.	.	.	.
Medium	Medium	Medium	Low	→	5
.	.	.	.	.	.
.	.	.	.	.	.
High	Medium	Medium	Medium	→	13
High	High	Medium	Medium	→	20
High	Medium	High	Medium	→	20
.	.	.	.	.	.
.	.	.	.	.	.
High	High	High	High	→	100

This chart is an example of how teams develop their view of “size.” The Team that developed this chart had specifics about what they thought size meant, including complexity and risk. I sanitized it for the article. So, for example, for this Team “risk” meant “known risks” while “uncertainty” was “unknown unknowns” (and there was more of this the further out the work was.) In their minds it was not the same thing. To be clear if your Team formalizes the development of a chart like this, their chart will look different based on the work, your Team’s make-up, etc.

What is really cool about estimating this way is that you will establish a stable Team velocity over time which will help the Team make and meet commitments. As you determine future work, you can rely on the historical velocity of the Team since the numbers already factor in different risk, complexity and uncertainty profiles.

Want to Know More?

- For more reasons for not equating story points with hours see
  - https://www.mountaingoatsoftware.com/blog/dont-equate-story-points-to-hours or
  - https://www.scruminc.com/story-points-why-are-they-better-than/

Team, Estimates, Forecast, FAQ, Points, EstimationApproach