

Scrum metrics

How and what scrum teams can measure to optimize team performance



BY ERIKA SA

BROWSE TOPICS

- Agile manifesto
- Scrum
 - Scrum master
 - Retrospectives
 - Distributed scrum
 - Roles
 - Scrum of scrums
 - Agile scrum artifacts
 - Scrum metrics
- Kanban
- Agile project management
- Product Management
- Agile at scale
- Software development
- Design
- The agile advantage
- DevOps
- Agile Teams
- Agile tutorials
- About the Agile Coach

All articles

Summary: Scrum metrics are specific data points that scrum teams track and use to improve efficiency and effectiveness. Scrum teams use metrics to inform decision-making and become more efficient in planning and execution, as well as set target goals and improvement plans.

"If you can't measure it, you can't improve it," noted famous management thinker Peter Drucker. While this isn't a statement that applies to every facet of life, it does apply to teams practicing agile scrum. By using certain metrics, scrum teams can adjust, pivot, and refine team effectiveness.

What is scrum?

Scrum is an agile framework and way of working that helps teams address complex problems, while iteratively developing solutions around a goal. The scrum way of working is characterized by the [sprint](#), which is a measured amount of time a scrum team works to complete a set amount of work.

Due to its adaptability, agile frameworks like scrum have spread beyond tech-oriented teams to support, design, marketing, and more. As such, scrum metrics are increasingly important to measure team performance and effectiveness.

What are scrum metrics?

Scrum metrics are specific data points that scrum teams track and use to improve efficiency and effectiveness. When defined, understood, and implemented, scrum metrics can become insights that help guide and improve a team's agile journey.

Scrum teams use metrics to inform decision-making and become more efficient in planning and execution. They can also be used to establish a baseline on the status quo and set target goals and improvement plans. When doing so, there is no industry standard yardstick to compare everyone against today. This is because comparing data

RELATED READING

Learn how to use sprints in Jira Software

[Learn more →](#)

SUBSCRIBE

Sign up for more articles

Email

[Subscribe](#)

points in the absence of context is like comparing apples and oysters. Every team is unique -- they can be different in size, technologies used, type of work they do, etc.

It's up to each team to agree on a set of metrics to track and define how to use them. This is not an individual effort and not something the leadership or management can define and enforce on behalf of the teams.

Why do you need scrum metrics?

Scrum metrics can help teams establish benchmarks and guide the direction of the work. For this reason, scrum metrics are helpful for established and new teams alike.

Tracking scrum metrics also helps bring visibility to various dimensions of a team's effectiveness, whether it's a team's velocity, capacity, predictability in delivery, or quality of the product. Key metrics can cultivate awareness in the team's performance and instigates action to change and improve. Plus, they can even help to gauge team happiness and satisfaction over time.

Often, it's too easy for many agile teams to rely on gut feelings or intuition to get a sense of a team's performance. While there can be many practical reasons behind this habit, it is nonetheless a big missed opportunity.

Can scrum metrics be used as KPIs?

Yes, and no. Scrum metrics can be used to set up key performance indicators (KPIs), but it depends on the type and scope of the work. Scrum metrics alone cannot measure customer value or show if the team even delivered the right thing. For an agile team, KPIs should ultimately show how well the team supports company priorities.

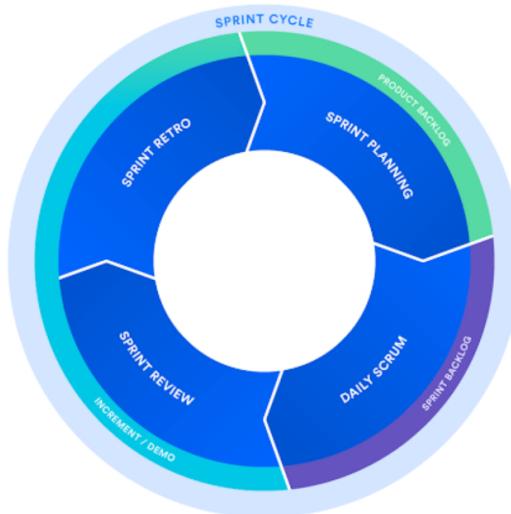
When measuring a scrum team's performance, other metrics outside those of the scrum should be considered, including:

- Return on investment (ROI) to a business – Companies measure this in numerous ways depending on the objectives, including growth in revenue, monthly active revenue (MAU), and more.
- Customer satisfaction – Survey metrics like the Net Promoter Score (NPS) and customer satisfaction score (CSAT) can help track the success of a project. Consistent customer satisfaction metrics for every release are important to showing a scrum team's value to customers.
- Team satisfaction – Just by asking your team about their level of motivation with the project and engagement with the team, you can catch problems like turnover,

attrition, and dissatisfied developers.

Key scrum events and what metrics to review

While agile scrum defines several recurring events – sprint, sprint planning, daily scrum, sprint review, sprint retrospective – these don't provide any guarantees of progress or success. However, each one allows team members to inspect and adapt how they work.



Sprint planning

A sprint planning meeting is held at the beginning of a sprint where a team breaks down story descriptions into detailed tasks. This provides an estimate of work to be produced during a sprint. There are a handful of data points that can make your team's sprint planning more efficient, including sprint goals, current team velocity, team capacity, and type of work. We use a [sprint planning meeting template](#) to help guide our sprint planning.

Sprint goals

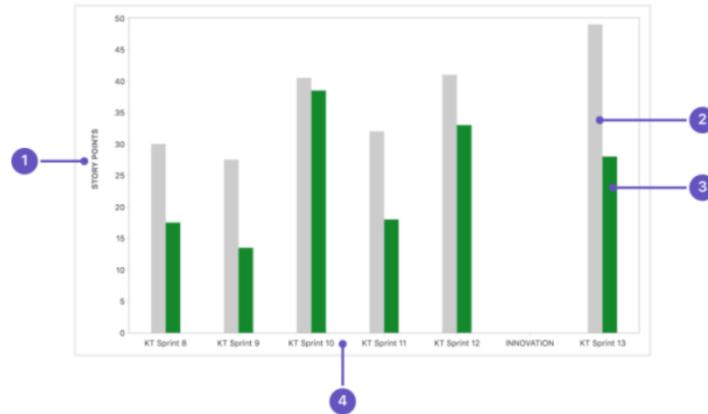
Sprint goals help teams decide what to accomplish in a sprint, bring cohesion to the items, and set priorities. Sprint goals often contribute to a bigger outcome than can be achieved through multiple sprints. The priority of the sprint goals should be based on their impact to this outcome. A truly effective team will review goals and their priorities regularly to strategize how to sequence and split engineering efforts.

Team velocity

How much a team can commit to a sprint essentially boils down to velocity, or how much work it can accomplish during a given time, as well as capacity, or how much availability it has. A velocity chart, like the [one we use in](#)

Jira, reveals the amount of value delivered during a sprint. This helps us to predict the volume of work a team can perform for future sprints. A team's velocity can only be understood after running a few sprints together as a team. Over time velocity will stabilize as a result of a team working together. This involves not only ramping up on technologies used but also understanding each other's expertise and learning how to work as a team.

The following is an [example of a velocity chart](#) with (1) estimation statistic based on story points, (2) commitment, which is an estimate of all issues in the sprint, (3) completed estimates, and (4) sprints completed.



Team capacity

It shouldn't be a surprise that the amount of work a team can complete in a sprint is based on its capacity and availability. Stable velocity won't mean anything if half of your team is off on a vacation. One way to plan capacity is to gather each team member's availability a few sprints out and add the sum to get a percentage of total capacity. Since last-minute changes or emergencies can happen to anyone, it's also a good idea to leave a 10 percent buffer in your sprint commitment to avoid overcommitting and under-delivering.

Type of work

When your sprint backlog is a growing mix of feature work, bug fixes, and technical debt, it becomes tricky to see where your team is dedicating their time in the sprint. It's easy for bugs or tech debts to sneak into your sprint, especially if the development team is passionate about quality. But if a team isn't careful, it can end up after the sprint wondering why it didn't ship enough customer values as planned.

Be intentional about what work your team is taking on by reviewing the split of different types of work during sprint planning. In this case, even if you find a lot of tech debt and quality work in the backlog, you can solve it strategically by scheduling a tech debt sprint or raising the bar on QA.

Stand-ups (aka daily scrum)

Stand-ups, or daily scrum, are short meetings held each day where team members check in about their work. For effective scrum teams, stand-ups should go beyond updates about an individual's updates on their to-do list. They are an opportunity to review a team's sprint progress and realign priorities for making small to big daily decisions that can have a significant impact on a sprint's outcome.

In doing so, the following data and metrics can come in handy:

Progress towards sprint goals

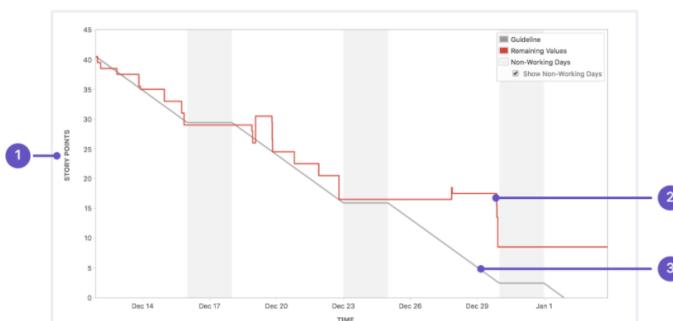
While team members might be clear about the status and progress of their work, it can be easy to miss the overall progress towards collective sprint goals. This is why it's important to bring up a list of sprint goals during a standup to review as a team.

Consider this a chance to discuss if the team is still on track. If not, why and what can be done about it? If it's something that can't be resolved, it's important to communicate this to key stakeholders, so everyone is on the same page.

Sprint burndown

To better understand a team's progress, your team should give a quick review of the sprint with a [sprint burndown chart](#). A sprint burndown chart tracks the completion of work throughout a sprint. It does so by comparing the time and amount of work to complete, measured in story points or hours. It helps predict a team's ability to complete work in a designated time and helps track scope creep. If a burndown chart has a sharp drop, it could be related to an inaccurate estimate of work.

The following is a sprint burndown chart in Jira, with (1) the estimation statistic, (2) the remaining values that are total amount of work left in the sprint, and (3) guideline that is an approximation of where your team should be.



Workload distribution

One thing that the team shouldn't lose sight of is how much work individuals are taking on. Especially in the remote working culture, it's hard to understand just how

In remote working culture, it's hard to understand just how much everyone is taking on. If you don't have visibility into this, it could mean some of your team members might be overworked. A standup is a place where team members ask each other for support and help; it can also be a place to adjust the workload of everyone to better meet sprint goals.

There is one thing to keep in mind when using this metric with your team -- don't turn it into a weapon. If you use this to measure an individual's productivity, you can end up discouraging your team. Instead, create a safe environment for everyone to openly talk about how much they're taking on, and where they need help.

Sprint retrospective

Even the best teams can benefit from [sprint retrospectives](#). This is when you and your team review what happened in the sprint by celebrating what went well, what needs improvement, and why. Naturally, this is the perfect time and place for you to review key sprint metrics, including sprint goal completion and sprint satisfaction.

The following is an example of my team's retrospective outlined on a Confluence page:

The screenshot shows a Confluence page titled "Retrospective". At the top, there is a video call interface with four participants. Below the video, there is a table with two columns: "What went well? (10 mins)" and "What needs improvement? (10 mins)".

What went well? (10 mins)	What needs improvement? (10 mins)
<ul style="list-style-type: none">• Real engineering as a cross-functional team with our Product and Marketing teams• Super strong collaboration on the latest release• We had really productive, but tough conversations we needed to have at the right time!• Strong alignment to our purpose and mission as a team• LOVED our project kickoff – super productive and energetic way to start a new year!• Sprint planning this week made it super clear what we needed to achieve and who was going to do what• Our latest release seems super useful – was really exciting to see it live• Really appreciate everyone's respect towards work/life boundaries	<ul style="list-style-type: none">• Priorities aren't super clear at the moment, which is challenging because we're getting so many requests for support• We don't know how to say no• Seems like we're facing some bottlenecks in our decision making• Still some unclear roles and responsibilities as a leadership team

At the bottom of the table, there is a section labeled "Action items".

Sprint goal completion

How did your team track against the goals set during sprint planning? If your team checked everything off the list, great! If not, what could have been better? In evaluating your team's success, scrum metrics we've already discussed can come in handy. Any improvement in your team's workflow is worth celebrating - maybe your team moved faster because there wasn't any scope creep. For teams practicing [DevOps](#), this can also be a place to review key DevOps metrics, such as cycle time or deployment frequency, to discuss improvements in the delivery process that can increase the likelihood of completing sprint goals. Doing so will help your team address the issue and come up with a clearer action plan.

Sprint satisfaction

This is simply asking your team about their satisfaction

with the sprint. Some teams use a number scale for this, anecdotal feedback, or even emojis/gifs. Your team can reflect on team goals and if the type of work was in alignment with team goals. Was an inordinate amount of time spent on tech debt versus finishing a feature?

Throughout the retrospective, encourage team members to speak up and call on them if needed. The best retrospectives have multiple perspectives and a variety of opinions. What's important is that by the end of the meeting, the team largely agrees on the top problems, owners, and a plan to follow up on key issues.

In conclusion...

The purpose of scrum is to help teams work better and the purpose of scrum metrics is to help teams ensure that