



Scrum: The Basics

Brought to you by Assemble You.

It's time to work on YOU. So sit back and listen to practical, actionable advice to accelerate your progress.

If you're a project manager, we don't need to tell you how much pressure comes with the job. It's up to you to lead the charge of a large-scale operation that will end up with a completed project that's satisfying for the stakeholders. A lot of money and company time is invested in your chances of success. You need to decide on a method that will guarantee that success.

Scrum is one such method. Scrum is one of several frameworks adhering to the Agile approach to project management. For those unfamiliar with it, Agile is an approach that breaks big projects down into manageable chunks called iterations, which are then worked on by cross-functional teams.

It differs greatly from traditional Waterfall management in that it's more collaborative and more adaptable. It delivers results in increments instead of just one final completed project, thus keeping stakeholders consistently in the loop. Scrum adopts many of the same principles but is its own unique entity.

Tim Stobierski of the Northeastern University explains the differences like this:

The key difference between Agile and Scrum is that while Agile is a project management philosophy that utilizes a core set of values or principles, Scrum is a specific Agile methodology that is used to facilitate a project.

Agile is a project management mindset with a collection of ideas, whereas Scrum is a specific way to put those ideas into action. Scrum has its own set of guidelines for the course of the workflow and each team member's role, which we'll get onto later.

Teamwork is one of the essential core values of Scrum, and it's where it gets its name from too. It's a rugby term, and it refers to a moment in a match where members of each

team will form a pack and engage in a battle of strength with the other team for possession of the ball. Every player in a scrum has their own important role, feeding into the ultimate goal of winning the ball.

It's a highly physical play, and success in a scrum is very much reliant on working together with the rest of the pack. The term was first used in relation to project management in a 1986 paper by Professors Hirotaka Takeuchi and Ikujiro Nonaka called *The New New Product Development Game*.

In today's fast-paced and fiercely competitive world of commercial new product development, speed and flexibility are essential. Companies are realizing that the old, sequential approach to developing products simply won't get the job done and are instead using a more holistic method — as in rugby, the ball is passed within the team as it moves as a unit up the field.

That was the mindset that drove the pioneers of Scrum and which is still in use today.

So that is what Scrum is. Let's now discuss how it works:

A Scrum project consists of two primary components: The team and the workflow. The team itself has clearly defined roles, the first of which is the **Product Owner**:

This position represents the voice of the stakeholders and the customer. They need to be an expert on the product and clearly understand what is expected of the team and the end result. It's a complex, demanding role that requires comprehensive knowledge and a broad skill set.

An analysis of Scrum Product Owners from *LucidChart* gave this description to the role:

One day they will need to access their deep well of market knowledge to strategize and present their vision to stakeholders, another day they will need to roll up their developer sleeves to help the team meet their goals during a sprint.

They are responsible for the Product Backlog, a list of requirements for the project ordered by priority. Product Owners must express this list to the rest of the team and order all necessary items. They must help the team optimize their workflow and keep all the relevant stakeholders informed on the project's progress.

The second role is the Development Team, which consists of a group of professionals executing the tasks required in the project. In the case of a software development project,

for which Scrum is a common approach, this team may consist of engineers, architects, developers, business analysts, testers, and any other relevant position.

They will be the ones delivering the project and handling the day-to-day duties, and they will work as a cross-functional team. There is no specific hierarchy within the Dev Team and no specific titles because all skills are necessary for project completion.

And the third role is what's known as the Scrum Master, whose job is to ensure that the rest of the team adheres to the Scrum approach. Like the Product Owner, it's a varied role, but it is more focused on guiding the team and keeping a close eye on their progress than ensuring stakeholders' satisfaction.

Max Rehkopf of the *Atlassian Agile Coach* outlined the position and listed its responsibilities as things like 'heading sprint planning meetings, one-to-one coaching, internal consulting, and facilitating sprint reviews'. Without the Scrum Master, the team's focus and motivation could start to spiral.

These specific roles all play their part in a streamlined daily process. Like in Agile methodology, the Scrum process takes an iterative approach, but it also divides each iteration into a series of 'events.'

All events receive a specific timebox that cannot be exceeded. That seems like it would be restrictive, but it's actually a surefire way to make your team more productive. You may have heard of Parkinson's Law, which states that "work expands to fill the time available for its completion," and this is true, but as Marc Zao-Sanders observed in a study for HBR, it has a subsequent effect. He states that:

We often spend more time on a task than we should, influenced by the time that happens to be available rather than how long the work should actually take. Disciplined timeboxing breaks us free of Parkinson's Law by imposing a sensible, finite time to a task and sticking to it.

Most of these timeboxed events center around "Sprints." The goal of a single Sprint is to produce something of value to deliver to the stakeholders for review. If the project is a software product of some sort, one Sprint might constitute the development of a single feature. If the project is building a house, you may dedicate one Sprint to one room.

As mentioned, a strict and specific amount of time will be afforded for the Sprint. In Ken Schwaber and Jeff Sutherland's 2020 version of their *Scrum Guide*, they recommend that it should never exceed the span of a month.

Sprint Planning is another event that is highly important in ensuring that the goals of a single Sprint are clear. For a Sprint that has been given a month to complete, the planning event will be timeboxed at roughly eight hours.

In this event, your team will discuss things like the purpose of the Sprint and its value to the overall project, what can be accomplished in the Sprint, and a clear plan of action. The planning event is also when the Product Backlog is analyzed and its contents clarified for the team.

Sprints are all concluded with two events, the first of which is a Sprint Review. During the Sprint Review, the results of the Sprint are given to the stakeholder, and a team discussion is held on how to proceed. The second event is the Sprint Retrospective, an opportunity to reflect on the previous Sprint and consider potential improvements moving forward.

There is also a unique event known as **the Daily Scrum**. Fifteen minutes at the same time every day, this is where the Development Team can discuss the previous day's achievements and what they hope to achieve during that day's Sprint.

Doing this every single day may seem insane, but Scrum is a process that relies on improvement by empiricism, and assessing your performance every day can help that. Willem-Jen Ageling put it best in his *Serious Scrum* series on *Medium* when he said:

It serves to assess how the activities you performed and the insights you gained in the last 24 hours impact the Sprint Goal and the Sprint Backlog, resulting in a plan for the next 24 hours to move closer to the Sprint Goal.

So while there are a lot of individual elements that go into a successful Scrum process, the result is actually quite simple to adapt to and will help you progress as a well-oiled machine.

To follow the Scrum method, you need to put together your Product Backlog, comprise a team consisting of a Product Owner, a Development Team, and a Scrum Master, and then proceed through the individual events that constitute a single Sprint.

As a product manager, if you want to truly maximize the potential of the experts you have at hand, then what you should take away from this is that a team is always stronger when working as a unit.

Gaining possession in a rugby match is achieved by the entire pack doing their share of the work in the scrum. You can apply that same line of thinking to find success with your project.

Thanks for listening, and have a great day.

Reading List:

- **[Article]** Agile vs. Scrum: What's the Difference, Tim Stobierski
- **[Article]** The New New Product Development Game, Hirotaka Takeuchi & Ikujiro Nonaka
- **[Article]** 7 Key Product Owner Responsibilities, Lucid Content Team
- **[Article]** What is a Scrum Master?, Max Rehkopf
- **[Article]** How Timeboxing Works and Why It Will Make You More Productive, Marc Zao-Sanders
- **[Article]** The 2020 Scrum Guide, Ken Schwaber & Jeff Sutherland
- **[Article]** The Daily Scrum? Why Daily? Are You Serious?, Willem-Jen Ageling