# Scrum for Everyone

## CERTIFICATION GUIDE

by Darcy Declute
@scrumtuous

Editors
Prachi Shaw
Amanda Punsammy

XENNIAL INNOVATIONS

# Scrum Certification Guide

[ ebook] | *_ebook.pdf*

# Chapter 0: Getting Started

I'm going to get you Scrum Certified. I hope you're ready.

The process is going to go something like this:

- We're going to go through the entire 2020 Scrum Guide together, one line at a time
- Together we'll extract as much meaning as possible out of the Scrum Guide's words
- We'll discuss how ideas in the Scrum Guide translate into exam questions
- We'll iteratively and incrementally test you on what you've learned

When you've finished this book, my goal is for you to be 100% ready to schedule and pass the Scrum Master Certification exam.

## The Best Ways to Learn

Before you read too far ahead, let me recommend that you download and print out a couple of copies of the 2020 Scrum Guide.

The Scrum Guide is only 13 pages long, and that includes the cover page, introduction and the table of contents. You won't be destroying a forest by having a couple of printed copies by your side.

*Figure 1. Try to hit as many learning modalities as you can whenever you learn something new. Just be sure to use high-quality, authoritative resources.*

Grab a highlighter as well and markup that printed copy as we go along. It'll help you learn.

## Read the Scrum Guide

Along with downloading it, you should also *read* the Scrum Guide before you dig into the first chapter of this book. In fact, read it a couple of times. It behooves you to do so.

I'm actually not a huge fan of the way the Scrum Guide is laid out. I'm a huge fan of the wisdom it contains. I'm just not a fan of the way it's structured.

For example, the first paragraph in the Scrum talks about the *Product Owner* and *Product Backlog items*. Yet the Scrum Guide itself doesn't explain what a Product Owner or a Product Backlog item is

for another five or six pages.

I don't like that. If you're going to bring up a topic, you should address it and explain it the first time it's mentioned. I don't like saying 'this is really, really important, but we'll cover that later.' If it's important, cover it now.

Having said that, this certification guide dissects the Scrum Guide, one line at a time, from the beginning to the end. That forces me to follow the layout of the guide, despite the fact that if it was up to me, I would have written it a bit differently.

So please, read the Scrum Guide a few times before jumping deep into this book. That way you'll have a better idea of what phrases like *the Sprint Goal* and *a usable Increment* mean.

With that foundation, we'll be able to dig much deeper into the significance of these interesting and important terms. Now let's get started with the Scrum Guide!


*Figure 2. Download and read the official Scrum Guide a few times before reading this book. Read it a few times after reading this book as well. I think you'll find the Scrum Guide makes a lot more sense after completing this study guide.*

## Chapter 1: What is Scrum?

How would you define Scrum?

To be successful on the Scrum Master Certification exam you have to commit to the Scrum Guide's definition of Scrum, which means abandoning the biases and misconceptions you may have adopted over years of hearing people talk about Scrum or seeing Scrum implemented in a less than pure manner.

Here's the 2020 Scrum Guide's first sentence. How well does this definition of Scrum work with the way you previously perceived it? (And I say 'previously', because this is the definition you must 100% commit to right now if you want to pass the Scrum Master certification exam.)

> Scrum is a lightweight framework that helps people, teams and organizations generate value through adaptive solutions for complex problems.
>
> — 2020 Scrum Guide page 3

Given what you know about Scrum, and taking into account any experiences you've had with Agile development, how would you rate this definition?

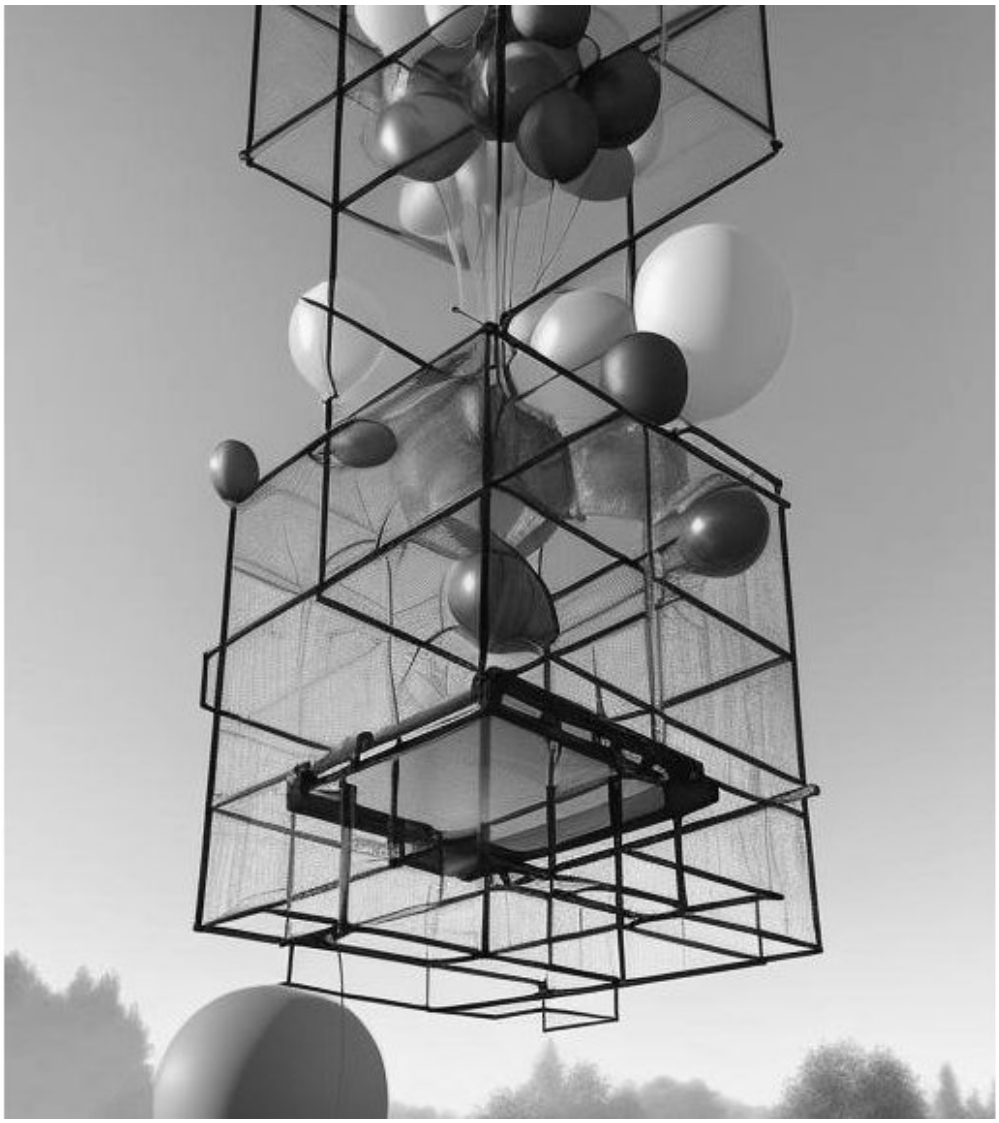


*Figure 3. Scrum is a lightweight, incomplete framework that helps teams solve complex problems as they work towards a Product Goal.*

## The Definition of Scrum

Whoever crafted that definition tried to make it as general and all-encompassing as possible, almost to the point where the definition doesn't provide much value.

- The term 'generate value' is very generic
- The term 'complex problems' could apply to anything
- The term 'adaptive solutions' sounds like marketing gibberish

But this is the definition we have, and this is the definition you will be tested on.

### The Words Not Spoken

The Scrum Guide say a lot in its brief 13 pages, but sometimes it's more interesting to focus on what the Scrum Guide *doesn't* say. Notice how:

- The official definition of Scrum never mentions software development
- The official definition calls Scrum a *framework,* not a process or methodology

Given the official definition of Scrum, how would you answer the following question?

---

### Test Yourself

Scrum is a proven software development process.

- ▢ True
- ▢ False

The answer is false. Scrum is not a process, nor does it specifically target software development.

You'll get beaten with a stick if any of the Scrum gatekeepers ever hear you call Scrum a process or a methodology. Scrum is a lightweight, incomplete framework.

- Scrum is not a process.
- Scrum is not a methodology.
- Scrum is purposefully incomplete.

Scrum doesn't try to solve all of your project management problems. It just helps you to get started, to keep going and to minimize risks along the way.

---

**TIP** It often helps to think about certification exam questions outside of the domain of software development. Reframe an exam question in terms of an isolated team building a shelter on a deserted island and answers may become more clear.

## Scrum is a Framework

Feel free to debate whether you believe Scrum is a process or a methodology on Twitter or in your favorite online forum. I know I have.

On the Scrum Certification exam? Scrum is a framework.

The stewards of the Scrum framework have also worked hard to position Scrum as a tool that can be applied in a variety of industries, not just software development.

If you ever see an option on the certification exam that asserts Scrum works exclusively in the domain of software development, avoid it, because it's wrong.

---

### Test Yourself

Here's the type of trick question you'll see on the Scrum certification exam that attempts to trip you up on the incorrectly held belief that Scrum is only used in software development:

> True or False: Scrum is a lightweight framework used exclusively by software development teams to generate value through adaptive solutions to complex problems.

The answer is false because the question implies that Scrum is only applicable in the world of software development.

There is a big push in the Scrum community to gain acceptance outside of software development. Any certification questions that pigeonhole Scrum into a software development box will be wrong.

---

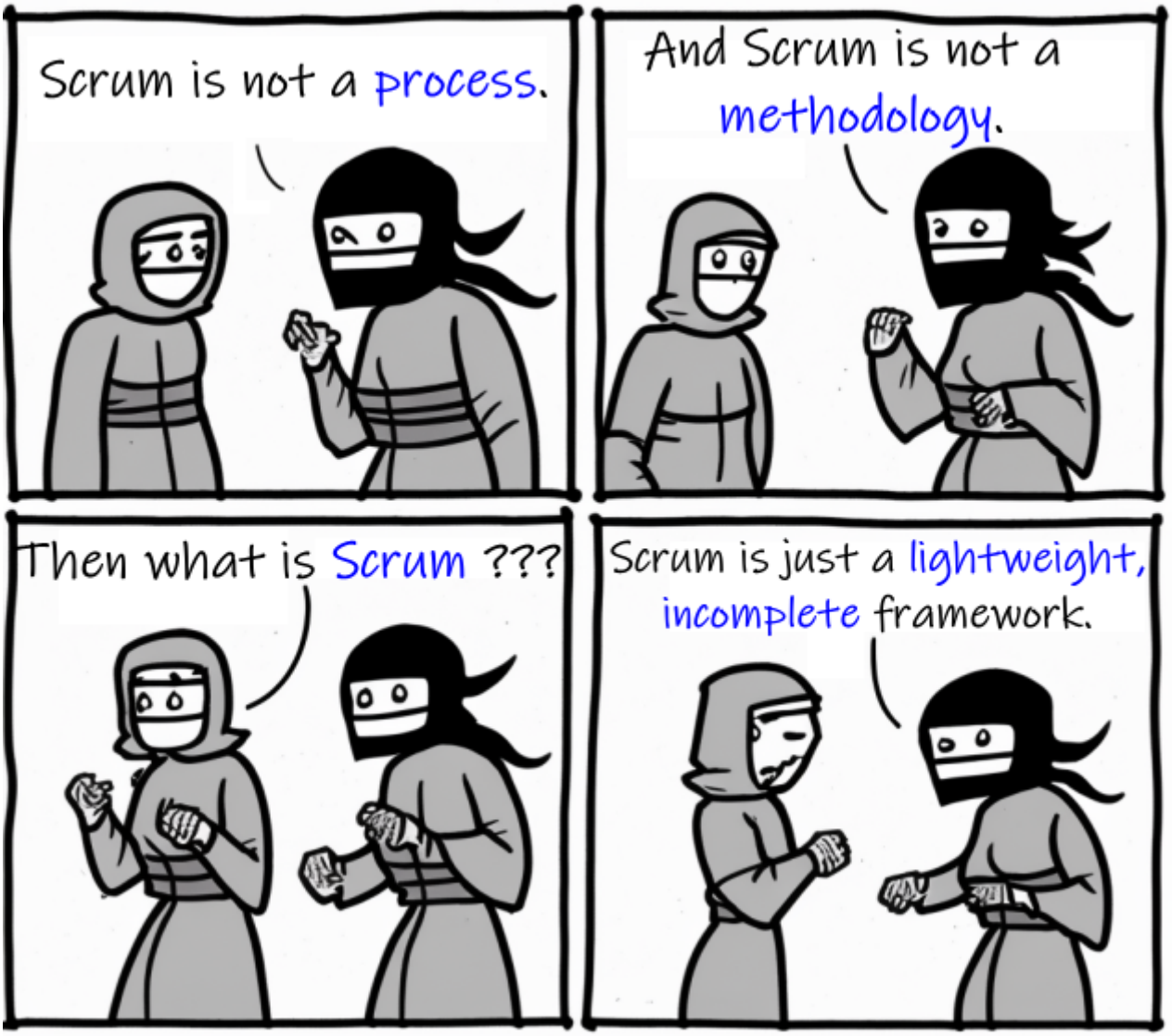
### Test Yourself

> Which one of the following statements most accurately reflects the definition of Scrum?
>
> - ▢ A) Scrum is a software development methodology
> - ▢ B) Scrum is an Agile process for teams and organizations to following
> - ▢ C) Scrum is a lightweight framework to help teams tackle complex problems
> - ▢ D) Scrum is a lightweight framework to help teams and organizations build software

Option C is correct.

The Guide describes Scrum as a "lightweight framework that helps people, teams, and organizations generate value through adaptive solutions for complex problems."

Any references to Scrum being a methodology, a process, or a framework that only targets software development will always be a wrong answer on the Scrum Certification exam.




## Iterative and Incremental

According to the Scrum Guide, here's a high-level overview of how Scrum is supposed to work.

> In a nutshell, Scrum requires a Scrum Master to foster an environment where:
>
> 1. A Product Owner orders the work for a complex problem into a Product Backlog.
> 2. The Scrum Team turns a selection of the work into an Increment of value during a Sprint.
> 3. The Scrum Team and its stakeholders inspect the results and adjust for the next Sprint.
> 4. Repeat
>
> — 2020 Scrum Guide page 3

The name 'Scrum Master' sounds intimidating.

People think that since the term 'master' is in the name, the Scrum Master controls everything.

The Scrum Master controls very little.

The Scrum Master's only real job is to coach people on how Scrum works, or as this paragraph states, 'foster an environment' where this iterative set of steps is performed.

## Scrum is Simple

People tend to overthink Scrum.

People think there are a bunch of rules they have to follow if they want to use Scrum. The fact is, there are very few rules in Scrum. The brevity of the Scrum Guide is proof of that.

Scrum is pretty simple, and when problems arise, it's pretty pragmatic too.

> Scrum is simple.
>
> Try it as is and determine if its philosophy, theory, and structure help to achieve goals and create value.
>
> The Scrum framework is purposefully incomplete, only defining the parts required to implement Scrum theory.
>
> Scrum is built upon the collective intelligence of the people using it.
>
> Rather than provide people with detailed instructions, the rules of Scrum guide their relationships and interactions.
>
> — 2020 Scrum Guide page 3

### Test Yourself

Which of the following statements are true about the Scrum framework?
(Choose 2)

- ▢ A) Scrum describes an iterative process
- ▢ B) Scrum is an iterative framework
- ▢ C) Scrum generates value by repeatedly delivering usable increments to the stakeholders
- ▢ D) Scrum only allows stakeholders to inspect progress when the final product is delivered

Options B and C are correct.

Scrum describes a set of steps that are to be repeated again and again. That makes Scrum *iterative*. But Scrum's an iterative *framework*, not an iterative *process*. So Option B is correct while Option A isn't.

Scrum is also an incremental framework, which means it constantly tries to deliver something tangible and of value to the client at the end of every sprint. That way the stakeholders can regularly give feedback. If there's an issue, the Scrum Team can then adapt.

That's in stark contrast to what is known as the Waterfall model where the client gets a complete product at the end of a long development cycle. So Option C is correct while Option D is wrong.

| | |
|---|---|
| **TIP** | Waterfall gets its name from the fact that isolated development phases like planning and design flow into each other, in only one direction, just like water in a waterfall. |

## It’s a Guide. It’s not an Instruction Manual

People often look to the Scrum Guide for definitive answers to things. The Scrum Guide doesn’t contain many definitive answers.

It’s a guide, not a rulebook.

The Scrum Guide even promises *not* to be heavy on rules, saying that it promises *not* to 'provide people with detailed instructions.'

There are very few actual rules in the 13-page Scrum Guide.

Outside of the few rules Scrum has, the framework encourages people to discover strategies that work best for them.

### Test Yourself

> True or False: Scrum is a complete and proven framework that helps teams achieve goals and create value.

This is false.

Scrum self-identifies as a *purposefully incomplete* framework.

This fact seems counter-intuitive to many. After all,

- Why would anyone want to use an incomplete framework?
- Wouldn’t a complete framework be better?

The incomplete nature of Scrum is actually what makes it so attractive. Scrum provides only enough direction to be useful, but not so much direction that it is restrictive. Scrum teams are given all the leeway they need to find the processes and frameworks that work best for them.

---

## Exposing Efficacy

One of the funny things about Scrum is that because it’s so simple, it can expose practices and processes that are wasteful and non-productive. It also allows developers to focus on the practices that make them most productive.

> Various processes, techniques, and methods can be employed within the framework.
>
> Scrum wraps around existing practices or renders them unnecessary.
>
> Scrum makes visible the relative efficacy of current management, environment, and work techniques so that improvements can be made.
>
> — 2020 Scrum Guide page 3

Since Scrum is a framework, not a process, other processes can be used within it.

## Combine the Scrum Framework with other Processes

For example, people often think Kanban is a competitor to Scrum, but there is nothing that says Scrum and Kanban can't be used together.

If you're not familiar with Kanban, don't worry. Kanban is never mentioned in the Scrum Guide, and for the Scrum Master certification exam, all you need to know is that it's an alternate development strategy.

### Test Yourself

> True or false: Scrum can be used alongside various processes and methodologies including Kanban and Lean.

This is true.

Scrum is not a process nor is it a methodology, and because of that, it can be used in conjunction with a variety of popular methodologies like Kanban and Lean.

The Scrum Certification Exam will not test you on the intricacies of Lean Manufacturing or Kanban. It's sufficient just to know that these are two processes commonly used in manufacturing and software development.

---

### Test Yourself

> True or False: When implemented properly, Scrum has the capacity to expose ineffective management.

This is true.

The iterative and incremental nature of Scrum, where constant inspection and adaptation is encouraged, will shine a light on ineffective practices that happen external to the Scrum Team.

That's what the Scrum Guide means when it says "Scrum makes visible the relative efficacy of current management, environment, and work techniques so that improvements can be made."

And with that question answered, we're done with the definition of Scrum.

Now on to a little overview of what Scrum theory is and what it's based on.

# Chapter 2: Scrum Theory

Scrum boasts of being built around the concepts of 'empiricism and lean thinking.'

The subject of 'Empiricism' is one of the 13 groups into which questions on the Scrum Master certification exam are categorized, so pay special attention to the concept.

> Scrum is founded on empiricism and lean thinking.
>
> Empiricism asserts that knowledge comes from experience and making decisions based on what is observed.
>
> Lean thinking reduces waste and focuses on the essentials.
>
> — 2020 Scrum Guide page 3

I'm not a big fan of the word empiricism.

Fundamentally, empiricism means using your six senses to understand the situation you're in. I'm not sure how my sense of smell or taste helps me as a Scrum Master.

In the context of Scrum, empiricism means understanding the situation you are in based on knowledge, experience, and verifiable facts.

Empiricism also ties in tightly with the Scrum Pillars of transparency, inspection, and adaptation. That is, if you can see exactly what is happening (transparency), you can honestly assess your progress (inspection) and based on this real knowledge, you can adapt.

Empirical analysis of your current situation is a more effective way to make decisions than following a plan written up six months ago, or making decisions based on trendlines on a historical chart.

## Empirical Example

Here's an example of the application of empiricism.

Imagine you have a burndown chart that says you are completing 10 items Sprint. How many items do you think you'll complete in the next Sprint?

You'd probably say 10, but that's an answer based on a chart. Charts don't have six senses. Charts may be predictive, but they're not empirical.

Now, say you know half the developers have taken time-off to be with their families during the next Sprint?

You'll probably reduce your projection by 50 percent. That decision is based on empirical evidence that a chart simply couldn't provide.

Lean thinking is also an important concept in Scrum. Avoid waste and embrace minimalist thinking when you're working on a project. That's *lean thinking*.

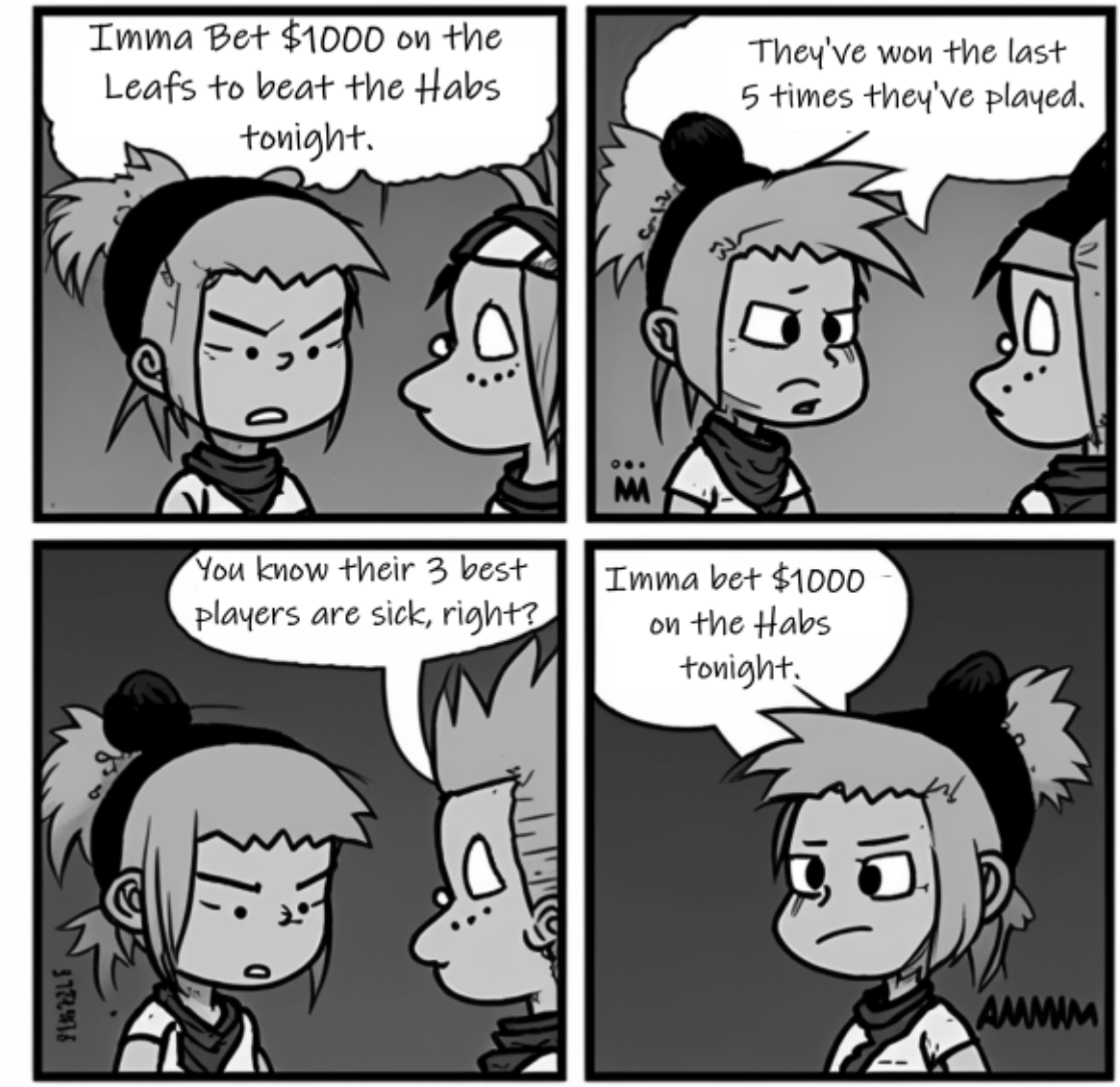



*Figure 4. Charts and trends are no replacement for first-hand knowledge and empiricism.*

## Empiricism and Lean Thinking

You are guaranteed to get two questions from this section on the Scrum Master certification exam. They are easy marks if you just understand what 'empiricism' and 'lean thinking' means.

### Test Yourself

Scrum theory emphasizes empiricism. That means a decision made by Scrum teams: (Choose 3)

- ▢ A) Should be based on factual evidence
- ▢ B) Should be based on intelligent assumptions
- ▢ C) Should be based on verifiable observations
- ▢ D) Should be driven by pragmatic speculation
- ▢ E) Should be based on experience

Options A, C, and E are correct.

Empiricism is all about using your experience, your gained knowledge, and verifiable observations (which is sorta the same as 'experience') to make decisions.

Concepts like 'speculation' and 'assumptions' run counter to the concept of empiricism, even if the speculation and assumptions are pragmatic and intelligent.

> "In empiricism, knowledge is spoken of as a posteriori, or "from the latter," meaning gained from experience. Simply put, empiricism is the idea that all learning comes from only experience and observations.
>
> The term empiricism comes from the Greek word for experience: empeiria. The theory of empiricism attempts to explain how human beings acquire knowledge and improve their conceptual understanding of the world."
>
> — TechTarget WhatIs Definition, Empiricism

---

## Test Yourself

Which of the following are characteristics of lean thinking?
(Choose 2)

- ▢ A) A focus on waste reduction and efficiency
- ▢ B) A command and control approach to problem-solving
- ▢ C) A minimalist approach that removes unnecessary steps
- ▢ D) A methodology that is made up of individualized, compartmental steps that flow into each other.

In this case, A and C are correct.

Efficiency, waste reduction, and the elimination of unnecessary steps within a process are all hallmarks of lean thinking.

The old Scrum Guide use to talk about the importance of *servant leadership.* That's still important, but it's been rephrased in the 2020 Scrum Guide. Now we talk about 'leaders who serve.'

The 'Command and Control' approach that the military takes is the opposite of 'servant leadership,' so option B is incorrect.

The last option describes the Waterfall methodology, which is the antithesis of the Scrum framework.

Empiricism and lean thinking lay the foundation for the Scrum Framework. Be comfortable with these two terms and understand their basic meaning to score a few easy marks on the Scrum Master certification exam.

## Predictability and Risk

> Scrum employs an iterative, incremental approach to optimize predictability and control risk.
>
> — 2020 Scrum Guide page 3

This one sentence is a lot to unpack.

Scrum is iterative because it describes a set of steps that get repeated over and over again. The iterative sequence of steps as outlined in the first section of the Scrum Guide are:

1. A Product Owner orders the work for a complex problem into a Product Backlog.
2. The Scrum Team turns a selection of the work into an Increment of value during a Sprint.
3. The Scrum Team and its stakeholders inspect the results and adjust for the next Sprint.
4. Repeat

Scrum is incremental. That's referenced in the second step of the iterative process:

"The Scrum Team turns a selection of the work into an Increment of value during a Sprint."

The idea of Scrum being incremental means that small victories, small units of value, and small pieces of the final product get created and added together slowly over time until the product is finally finished.

Piece by piece, through the delivery of value added upon value, the product gets built. That's the incremental process.

By getting these small increments into the hands of stakeholders, and getting immediate feedback from which the Scrum Team can adapt, Scrum reduces risk and allows developers to better service the needs of their clients.

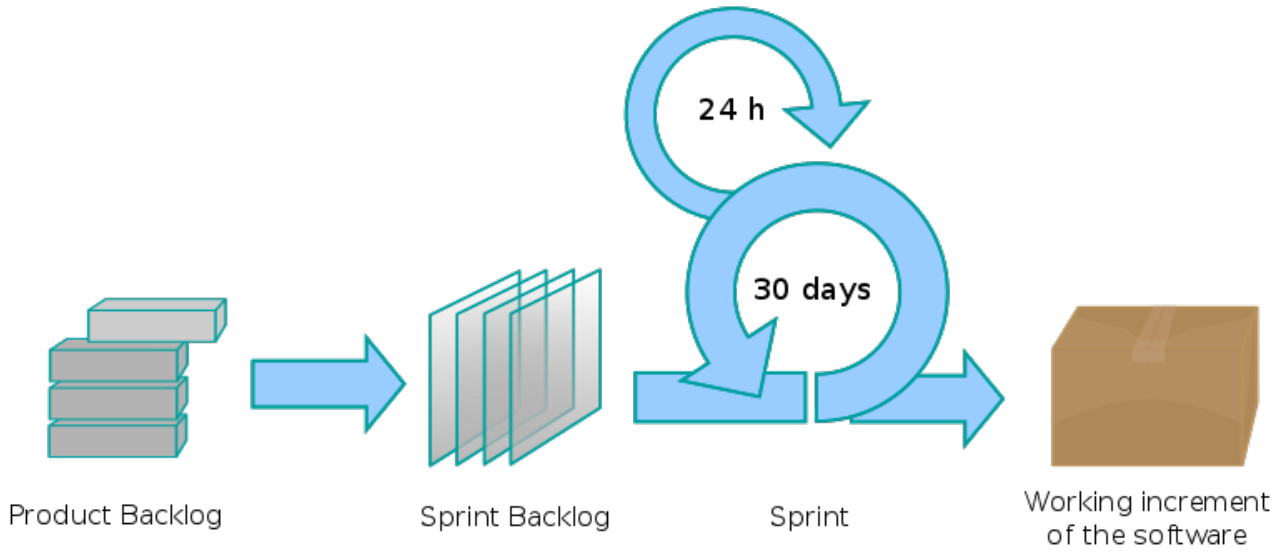



*Figure 5. Scrum development is iterative and each Sprint brings incremental progress. (Author: Lakeworks, GFDL1.2)*

## Cross-Functional Teams

The idea that teams should be cross-functional and self-managed is a key concept in Scrum, and it's one that you'll be tested on multiple times when you sit for the Scrum Certification exam.

> Scrum engages groups of people who collectively have all the skills and expertise to do the work and share or acquire such skills as needed.
>
> — 2020 Scrum Guide page 3

This is another extremely loaded statement. Understanding this paragraph will go a long way towards properly answering some of the most challenging questions on the Scrum Master certification exam.

Scrum assumes that the Scrum Team has all the skills required to build the product being developed.

- Does your project need testers? Then those people are on the Scrum team.
- Does your project need an architect? That person is on the Scrum team.
- Does your project need a performance or security specialist? Then a person with those skills must be on the Scrum team.

And what if your Scrum team doesn't have those skills? Then your team acquires them.

### Test Yourself

One of the items under development as part of your project is a spaceship to Mars, but nobody on your team knows how to build a spaceship to Mars.

Which one of the following options is the best way for the team to move forward?

- ▢ A) Outsource the development of a spaceship to a third party
- ▢ B) Remove the development of a spaceship from the project's requirements
- ▢ C) Explain to the Product Owner that you don't have the skills to build a spaceship to Mars
- ▢ D) Get the team to start learning about how to build a spaceship to Mars

Option D is correct.

This question is silly to the extreme, but it makes a point.

According to Scrum, all of the skills required to build a project under development exist on your team, or your team will take it upon themselves to acquire the skills needed. If your team outsources work to a third party, then the work in question is no longer within the control of the team, which means it's no longer part of the Scrum process.

That's what the Scrum Guide means when it says "Scrum engages groups of people who collectively have all the skills and expertise to do the work and share or acquire such skills as needed."

## The All-Encompassing Sprint

> Scrum combines four formal events for inspection and adaptation within a containing event, the Sprint.
>
> — 2020 Scrum Guide page 3

This statement is the source of the most commonly asked trick questions on the Scrum Certification exam, which are:

- Which events happen after the Sprint finishes?
- Which events happen before a Sprint begins?
- When a Sprint ends, when does the next Sprint begin?

Scrum has four time-boxed events that happen within a fifth Scrum event known as a Sprint. Sprint Planning, the Daily Scrum, the Sprint Review, and the Sprint Retrospective all happen within the confines of a Sprint.

- None of the Scrum events happen after a Sprint
- None of the Scrum events happen before a Sprint.
- None of the Scrum events can be left out of a Sprint.

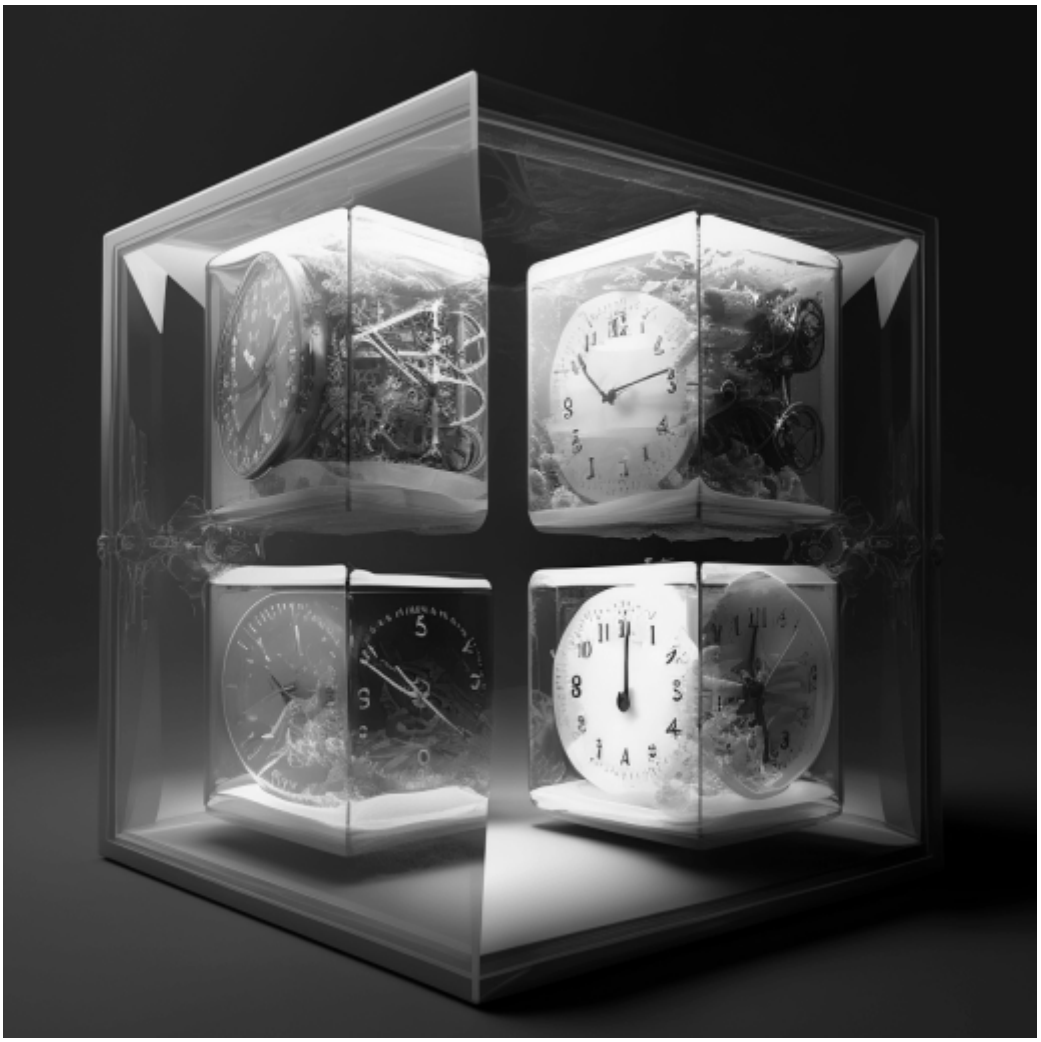


*Figure 6. The four timeboxed events in Scrum all occur inside a fifth event known as the Sprint.*

## Moving from Sprint to Sprint

Everything in Scrum happens within the boundaries of a Sprint. As soon as one Sprint ends, the next Sprint begins.

There is no buffer time between when one Sprint ends and the next Sprint starts where integration takes place, quality assurance happens or testing is done. If any of those things are part of the development of your product, all of those things have to happen during the Sprint.

Don't get tripped up on any questions that ask what happens before or after a Sprint.

### Inspection and Adaption

Notice how the Scrum Guide states that the higher purpose of the different Scrum Events, such as the Review, Retrospective, Planning meeting, and the Daily Scrum is to 'inspect and adapt.'

> Scrum combines four formal events for inspection and adaptation within a containing event, the Sprint.
>
> — 2020 Scrum Guide page 3

You will often get questions on the Scrum certification exam about what the purpose of the Sprint Retrospective is or what the purpose of the Daily Scrum is. If any of the listed options include the terms 'inspect' or 'adapt', those will likely be the correct answers.

### Test Yourself

Which one of the following options best describes when a new Sprint begins?

- ▢ A) A new Sprint begins when Sprint Planning is completed
- ▢ B) A new Sprint begins when the Sprint Review is completed
- ▢ C) A new Sprint begins when the Product Owner begins the Sprint in JIRA
- ▢ D) A new Sprint begins after Sprint Planning when the Scrum Master declares the start of the Sprint
- ▢ E) A new Sprint begins as soon as the previous Sprint ends

Option E is correct. A new Sprint begins as soon as the previous Sprint ends.

Technically speaking, the last event in the Sprint is the Sprint Retrospective, and the end of this event officially marks the end of the Sprint.

If we were to temporarily jump ahead 7 pages in the Scrum Guide, we'd see that stated in plain text.

> The Sprint Retrospective concludes the Sprint.
>
> — 2020 Scrum Guide page 10

## Test Yourself

What is the purpose of the Daily Scrum?
(Choose 2)

- ▢ A) For the Scrum Master to get daily status updates from the developers
- ▢ B) To allow the developers to inspect their progress toward the Sprint Goal
- ▢ C) For the Product Owner to track the development team's progress on Product Backlog items
- ▢ D) To allow the developers to adapt their Sprint Plan as they work towards the Sprint Goal

Options B and D are correct.

From day to day and hour to hour, conditions change.

Scrum recognizes this reality, which is why it provides several events that allow for the inspection of progress along with the ability to adapt if necessary.

It should be noted that inspection and adaptation can happen at any time during the Sprint, not just during the official Scrum events.

If a computer hosting the Git repo catches fire, you don't wait until tomorrow's Daily Scrum to put it out, nor would you wait to tell the rest of the team about it.

# Chapter 3: Scrum Pillars and Values

'Scrum values' is one of the 13 categories upon which the Scrum Master certification exam is graded.

The 'Scrum Pillars' don't constitute a specific category, but they frequently weave their way into answers about why a given event is taking place, or why a given Scrum artifact is important.

Memorize the values and pay close attention to the pillars. They are key to passing the Scrum Master certification exam.

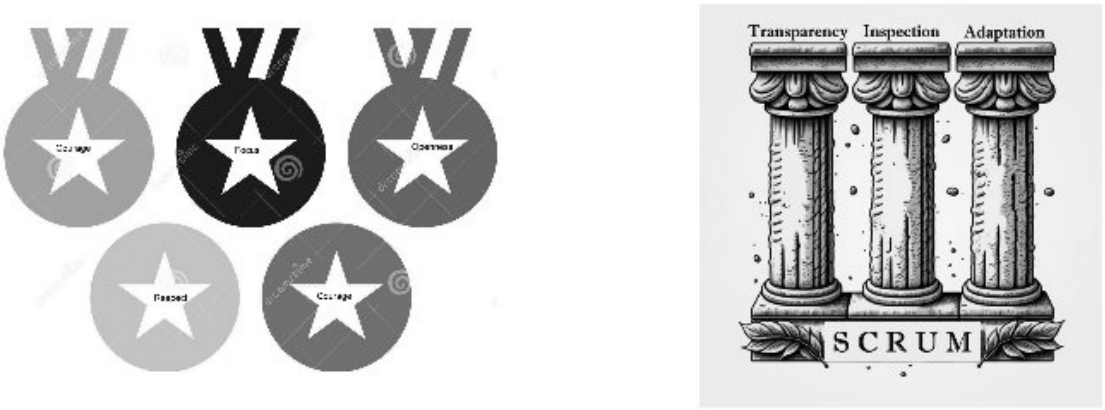



*Figure 7. The five Scrum values are: commitment, focus, openness, respect and courage. The Scrum pillars are transparency, inspection and adaptation.*

## The Scrum Pillars

We've already seen references to two of the Scrum Pillars in our discussion of the Scrum Guide:

> Scrum combines four formal events for inspection and adaptation within a contained event, the Sprint.
>
> — 2020 Scrum Guide page 3

Here's the Scrum Guide's formal declaration of the three of them:

> These events work because they implement the empirical Scrum pillars of transparency, inspection, and adaptation.
>
> — 2020 Scrum Guide page 3

It can't be emphasized enough. The three pillars of Scrum are:

- Transparency
- Inspection
- Adaptation

Everything that happens in Scrum harks back to the values and pillars.

The question as to why a particular event happens or why a particular artifact is required typically refers back to the Scrum pillars, with the Scrum values providing further support.

## Scrum Values

The five Scrum values that support the Scrum Pillars are:

- **Commitment:** The idea that a Scrum Team is made up of conscientious and motivated developers who are dedicated to achieving the Product Goal.
- **Focus:** The idea that the Scrum Team concentrates on the work at hand and doesn't plan too far into the future.
- **Openness:** The idea that developers work in a psychologically safe space where people can voice their opinion and share ideas without fear of reprisal.
- **Respect:** The idea that you will treat people with dignity and compassion, while doing the same in turn.
- **Courage:** The idea that a member of the Scrum Team will speak out against unethical practices or behaviors that put the project at risk.

Memorize the Scrum pillars and values, and don't confuse them with each other.

### Test Yourself

Which three of the following are Scrum pillars?

- ▢ A) Commitment
- ▢ B) Inspection
- ▢ C) Openness
- ▢ D) Focus
- ▢ E) Adaptation
- ▢ F) Respect
- ▢ G) Transparency

Options B, E and G are correct.

The Scrum pillars are transparency, inspection, and adaptation.

---

## The Endless Feedback Loop

The Scrum Pillars feed into each other. It's a feedback loop that drives the entire Scrum framework.

Transparency allows all participants to honestly inspect progress.

Inspection can reveal flaws in the plan. This leads to adaptation.

As teams adapt, they must be open about how well the adaptation has worked. This requires the team to be transparent, which then leads to inspection and adaptation. It's a constant loop.

## Inspection

> The Scrum artifacts and the progress toward agreed goals must be inspected frequently and diligently to detect potentially undesirable variances or problems.
>
> To help with inspection, Scrum provides cadence in the form of its five events.
>
> An inspection enables adaptation. Inspection without adaptation is considered pointless.
>
> Scrum events are designed to provoke change.
>
> — 2020 Scrum Guide page 4

Inspection is all about taking a look at what you and your team is doing so you can figure out if it's working or not.

- Why do we do the Daily Scrum? We do it so the developers can *inspect* their work and change their plans if they need to.
- Why do we do the Sprint Review? We do it so the team and the stakeholders can *inspect* what's been done and see if it's good or not.
- Why do we do the Sprint Retrospective? We do it to *inspect* how well the team worked together during the Sprint.
- Why do we have Scrum Artifacts? So people can *inspect* them and use them as a basis for adaptation.

The need for inspection throughout a Sprint is a common justification for Scrum events and artifacts. Events and artifacts enable the feedback loop that is built upon the Scrum Pillars.

### Test Yourself

Inspection in Scrum is: (Choose 2)

- ▢ A) Done frequently to detect potential problems
- ▢ B) Done infrequently to allow teams to concentrate on development
- ▢ C) Done to identify undesirable variances
- ▢ D) Done to ensure undesirable variances do not occur

Options A) and C) are correct.

Inspection is done frequently to detect potential problems and to identify any undesirable variances so that the team can address them.

The concept that anything could ever be done to ensure variances never occur, as option D suggests, is just wishful thinking.

In the real world, undesirable variances will always occur. Scrum recognizes that reality and helps teams deal with them early and adapt.

## Transparency

To know exactly how product development is progressing, everyone must be transparent about the work they are doing.

Empiricism doesn't work if we cannot look transparently into the results of the processes and methods we use within the Scrum framework.

> The emergent process and work must be visible to those performing the work as well as those receiving the work.
>
> With Scrum, important decisions are based on the perceived state of its three formal artifacts.
>
> Artifacts that have low transparency can lead to decisions that diminish value and increase risk.
>
> Transparency enables inspection. Inspection without transparency is misleading and wasteful.
>
> — 2020 Scrum Guide page 3

One of the Scrum values is openness. One of the ways to be open is to be transparent about the work being done and transparent about the progress being made.

The Sprint Backlog creates transparency because it lists everything the Scrum Team is working on, what the team's goal is during this Sprint, and the team's plan for achieving that goal. If stakeholders want to know what the Scrum Team is working on, they can look at the Sprint Backlog. It provides transparency.

What is the quality standard the team is using? Transparency into that is provided by the team's Definition of Done.

What will the team try to build next? Transparency into that is provided by the way the Product Backlog is prioritized.

Like inspection, the Scrum pillar of transparency is woven into all of the Scrum events and artifacts.

### Test Yourself

Which of the following problems will occur if product development lacks transparency? (Choose 1)

- ▢ A) Technical debt will embed itself in deliverables
- ▢ B) Product delivery dates will be difficult to estimate
- ▢ C) Stakeholders will begin asking for status updates
- ▢ D) Honest inspection of the Scrum Team's progress becomes impossible.
- ▢ E) All of the above

Option E) is correct.

Without transparency, it is impossible to know exactly what is happening throughout the development process. This destroys trust, makes honest inspection impossible and prompts management to want to micro-manage developers.

The state of progress of the Scrum Team must be transparent to all, otherwise the team and the stakeholders don't have the facts they need to make the right decisions for the future of the product.

---

## Adaptation

Things never go according to plan. That's one of the reasons we don't spend months planning things in Scrum.

It's more important to produce something of value than it is to waste time planning because nothing ever goes according to plan.

When plans do go awry, Scrum developers adapt. That's how they achieve their goals.

> If any aspects of a process deviate outside acceptable limits or if the resulting product is unacceptable, the process being applied or the materials being produced must be adjusted.
>
> The adjustment must be made as soon as possible to minimize further deviation.
>
> Adaptation becomes more difficult when the people involved are not empowered or self-managing.
>
> A Scrum Team is expected to adapt the moment it learns anything new through inspection.
>
> — 2020 Scrum Guide page 4

It's understood that in the world of software development, things change quickly.

Things also change quickly in the field of construction, manufacturing, banking, etc.

The ability for teams to quickly adjust and change their plan when things go sideways is a core tenant of Scrum. It's one of the reasons we have the Daily Scrum - it allows developers to collectively discuss problems and adapt as needed.

### Always Adapt

The Scrum Guide describes several artifacts and time-boxed events that provide an opportunity to adapt. However, these are not the only times the team is allowed to meet, speak, and adapt.

If a problem comes up during the day, a Developer doesn't have to wait until the next day's Daily Scrum to adapt. Nor does a Developer have to wait until the next day's Daily Scrum to discuss issues with fellow developers.

If a problem arises, teams are encouraged to inspect and adapt immediately.

If the team's war room catches fire, don't wait until tomorrow's Daily Scrum to leave the building. Ongoing inspection and adaption are requirements in Scrum.

---

## Test Yourself

If the process used to track development throughout the Sprint deviates outside of an acceptable limit, the Scrum development team should:

- ▢ A) End the Sprint early and begin a new round of Sprint Planning
- ▢ B) End the Sprint early and do a Sprint Review with all stakeholders
- ▢ C) End the Sprint early and do an internal Sprint Retrospective to see what went wrong
- ▢ D) Adapt during the Sprint and continue to push towards the Sprint goal.

Option D is correct.

Scrum is all about adaptation. If things don't go according to plan, the team should adapt. They certainly shouldn't end the Sprint. For the most part, Scrum doesn't allow them to.

- The Scrum Developers cannot end a Sprint early in Scrum.
- The Scrum Master cannot end a Sprint early in Scrum.

Only the Product Owner can do that, and only under the very special condition in which the Sprint Goal has become obsolete.

If things go sideways during a Sprint, the solution is not to cancel the Sprint or end the Sprint early. The solution is to adapt and continue to work towards the Sprint goal.

Sprints are short, typically between 2 to 4 weeks. Even if things go completely sideways, it won't be too long before a new Sprint begins, so continue to work hard toward the Sprint Goal. A new Sprint is always just around the corner.

---

## Test Yourself

A serious security-related bug has appeared in the code written by a fellow Developer and you need more details about the problem to fix it. When should this issue be discussed with the Developer?

- ▢ A) When the Scrum Master can coordinate a meeting between the two of you
- ▢ B) After the Quality Assurance (QA) team has time to investigate
- ▢ C) During the next scheduled Daily Scrum
- ▢ D) You should go over to the Developer's desk and discuss it now

Option D is correct.

There are scheduled invents in Scrum that provide opportunities to inspect and adapt, but those should never be used to limit communication and interaction between members of the team.

If a problem arises in Scrum, there's no requirement to wait until a Scrum event happens to address it. Address problems immediately and adapt.

## Scrum Values

Scrum is a simple, incomplete framework that doesn't solve every possible problem a development team will encounter.

What Scrum does do is provide five values it believes are important. When problems arise, the best solutions will respect these five values.

> Successful use of Scrum depends on people becoming more proficient in living five values:
>
> - Commitment
> - Focus
> - Openness
> - Respect
> - Courage
>
> The Scrum Team commits to achieving its goals and to supporting each other.
>
> Their primary focus is on the work of Sprint to make the best possible progress toward these goals.
>
> The Scrum Team and its stakeholders are open about the work and the challenges.
>
> Scrum Team members respect each other to be capable, and independent people are respected as such by the people with whom they work.
>
> The Scrum Team members dare to do the right thing to work on tough problems.
>
> These values give direction to the Scrum Team about their work, actions, and behavior.
>
> The decisions that are made, the steps taken, and the way Scrum is used should reinforce these values, not diminish or undermine them.
>
> The Scrum Team members learn and explore the values as they work with the Scrum events and artifacts.
>
> When these values are embodied by the Scrum Team and the people they work with, the empirical Scrum pillars of transparency, inspection, and adaptation come to life building trust.
>
> — 2020 Scrum Guide page 10

Like the Scrum pillars, the Scrum values help to justify why we perform the Scrum events and why we create the Scrum artifacts.

Quite often when a Scrum Master, Product Owner, or Scrum Developer is faced with a difficult challenge, the answer to the problem lies in how to conjure up a solution that is in line with these Scrum values.

For the exam, know the Scrum values and how each of them is defined. There is usually a question or two that will test to see if you know what the Scrum values are and what they mean.

## Test Yourself

According to the Scrum Guide, which of the following is **not** a Scrum Value? (Choose 5)

- ▢ A) Agreeableness
- ▢ B) Commitment
- ▢ C) Conscientiousness
- ▢ E) Openness
- ▢ E) Respect
- ▢ F) Extroversion
- ▢ G) Focus
- ▢ H) Emotional stability
- ▢ I) Courage
- ▢ J) Honesty

Options A), C), F), H), and J) are correct.

The five Scrum values are Commitment, Focus, Openness, Respect, and Courage.

Honesty is not one of them, but that's not to say you shouldn't be honest. Always be honest!

The other traits, extraversion (also often spelled extroversion), agreeableness, openness, emotional stability (neuroticism), and conscientiousness are together known as the Big 5 personality traits.

The big 5 personality traits are worth looking into if you're into psychology and human behavior, but you won't be tested on them when you take the Scrum Master certification exam.

# Chapter 4: Scrum Teams

Two of the thirteen categories into which Scrum Certification exam questions are grouped include:

- Scrum Teams
- Self-Managing Teams

Pay close attention to how Scrum Teams are structured, the role self-management and self-organization has, and avoid clouding the simple approach Scrum takes to team dynamics with what you may have learned in environments where Scrum may not have been applied properly.

## Composition of a Scrum Team

> The fundamental unit of Scrum is a small team of people, a Scrum Team.
>
> The Scrum Team consists of one Scrum Master, one Product Owner, and Developers.
>
> — 2020 Scrum Guide page 5

Take note of the fact that a Scrum team is made up of three *things:* one Scrum Master, one Product Owner, and multiple Developers.

Notice how I called them 'things.'

People like to call them 'roles.' The Scrum Guide never uses the term 'role' once. Technically, these are not roles.

The Scrum Guide later refers to them as 'accountabilities.' They are technically not 'roles.'

*Figure 8. A Scrum Teams has one Product Owner, one Scrum Master and developers. What does your Scrum Team look like?*

## Test Yourself

Here's a typical Scrum Certification exam question that tests your knowledge of the proper structure of a Scrum Team.

> Which one of the following best describes the typical composition of a Scrum Team?
>
> - ▢ A) A Scrum Team is made up of one Scrum Master, many Product Owners, and many Developers
> - ▢ B) A Scrum Team is made up of one Scrum Master, one Product Owner, and many Developers
> - ▢ C) A Scrum Team is made up of many Scrum Masters, many Project Managers, and many Developers
> - ▢ D) A Scrum Team is made up of one Scrum Master, one Project Manager, one Product Owner, and many Developers

The answer is B.

A Scrum Team is made up of one Scrum Master, one Product Owner, and many Developers

You will often see the term *Project Manager* appear in certification questions to throw you off. There is no *Project Manager* in Scrum. There is a Product Owner, but there is no such thing as a Project Manager in Scrum.

There are also no:

- Testers
- Architects
- Engineers
- Plumbers
- Carpenters
- Rocket scientists or
- Hairdressers referenced in the Scrum Guide.

If people with those skills are required to build the product you're working on, then they all fall under the term 'Developer.' There are no separate *Quality Assurance (QA)* people or *Business Analysts* in Scrum. There are no separate testing teams.

If your project needs testers, blacksmiths, or QA people, then they are considered developers on the team.

If you ever see references to *Quality Assurance Teams* or *Project Managers* or *Tester* roles as options on the Scrum Master certification exam, those options are typically incorrect.

### Test Yourself

The software developers on the Scrum Team are frustrated that their time is being consumed by tasks like user documentation and user acceptance testing. They would prefer to concentrate on core development activities.

Which one of the following options best describes how a team of software developers should operate under the Scrum framework? (Choose 1)

- ▢ A) Software developers should do as much development as they can and then pass quality control work off to QA people.
- ▢ B) Software developers should only build core features and then allow others to take care of non-functional aspects.
- ▢ C) Software developers should outsource work they are not skilled to perform to third parties or external teams.
- ▢ D) All work should be completed by the developers on a Scrum Team, including testing and documentation.

Option D is correct.

This question is a typical example of a Scrum Master certification question describing how many software development teams who say they are doing Scrum behave in an attempt to get you to incorrectly answer the question.

Many Scrum Master certification exam questions will talk about professions like QAs, business analysts, or engineers who are not on the Scrum team. Anyone doing work as part of product development must be *on* a Scrum Team. Regardless of what their skills are, they are simply seen as a Developer in the eyes of the Scrum Guide.

Options on the Scrum Certification exam where they start talking about specific professions are usually wrong because a Scrum Team only has three accountabilities: one Scrum Master, one Product Owner, and Scrum Developers.

---

## Subteams are Radioactive in Scrum

In Scrum, the development team is a collection of 'equals', regardless of their skills or experience.

In Scrum Team, there are no separate 'QA Teams' or 'Testing Teams', nor is there even a team lead.

Everyone pitches in to do the required work, and with the team's cross-functional skills, they can accomplish any specified testing or QA requirements.

> Within a Scrum Team, there are no subteams or hierarchies. It is a cohesive unit of professionals focused on one objective at a time, the Product Goal.
>
> — 2020 Scrum Guide page 5

The topic of subteams and hierarchies will be at the heart of seven or eight questions on the 80 question Scrum Master certification exam. They're easy to get right if you just remember that there are no sub-teams and there are no hierarchies in Scrum.

What does it mean to say there are no subteams in Scrum?

- It means there is no QA team to check for quality issues. The development team does that.
- It means there is no testing team to check for bugs. The development team does that.
- It means here is no performance team to do load testing. The development team does that.

If bugs need to be fixed, if quality needs to be assured, or if performance needs to be tested, then the people doing that work are part of the development team. To be without hierarchies means developers aren't given titles like 'Team Lead' or 'Sr. Developer.' Everyone on the team is equal.

## Test Yourself

Here's the structure of a prototypical Scrum Master certification exam question that tests you on this hierarchies and subteams:

There is an issue. What should you do? (Choose 1)

- ▢ A) Talk to the Senior Developer on the Scrum Team about it
- ▢ B) Talk to someone on the QA team about it
- ▢ C) Talk to someone on the testing team about it
- ▢ D) Coach the developers around strategies of how to address it as a team

Option D is correct.

You should immediately identify the first three options as being wrong because they talk about a Senior Developer (hierarchy), a QA team (subteam) and a testing team (subteam).

Scrum doesn't have subteams or hierarchies. If there's ever an issue that falls on the shoulders of the Scrum Master to solve, the solution is typically for the Scrum Master coach around it.


*Figure 9. The only time you have a subteam in Scrum is if there is literally a team of developers working on a sub. (Credit: USS Albacore, US Navy)*

## Cross-functional Scrum Teams

The importance of the concept of cross-functional teams that can accomplish everything required of them without resorting to third parties our outsourced help can't be emphasized enough for people who wish to pass the Scrum Certification exam.

> Scrum Teams are cross-functional, meaning the members have all the skills necessary to create value for each Sprint.
>
> — 2020 Scrum Guide page 5

The cross-functional aspect of a Scrum Team confuses a lot of test takers.

If you bake a cake, you need all the ingredients before you begin, right?

And if you want to build a product, you need a team of developers who have all of the skills needed to build that product.

If your team is going to build, test, secure and load test an application, then your team has to have developers on it with all of those skills.

Does the product under development require a warp drive made from dilithium crystals? Then your cross-functional team better have someone with the skills to build that, or at the very least, they better be actively learning how to do it.

A Scrum team doesn't outsource their work, nor do they hand their work over to a 'testing team' or a 'quality assurance' team. The Scrum Team itself is cross-functional and capable of doing whatever is required to produce the product under development. That's what it means to be cross-functional.

### Test Yourself

The next feature required by the Product Owner is a Mars orbiter, but nobody on your team knows how to build a Mars orbiter. What should you, the Scrum Master, advise the team to do? (Choose 1)

- ▢ A) Tell the dev team the Mars Orbiter will be removed from the required features list.
- ▢ B) Cancel the Sprint until the team finds someone who can build a Mars orbiter.
- ▢ C) Tell the dev team the Mars Orbiter feature will be outsourced to Elon Musk and SpaceX.
- ▢ D) Coach the dev team on the importance of learning and acquiring the skills required to complete all of the required product features.

Option D is correct.

This may sound like a silly question, but what would happen if you were on a team that needed to build a Mars Orbiter? You'd have to research how to do it and figure it out. And if you were hired to build a Mars Orbiter, the assumption is that you are indeed someone smart enough to figure out how to solve that impossible problem.

## Cross-Functional and Self-managed

> In Scrum, the developers are assumed to be cross-functional, and as a group, they will possess all of the skills needed to produce the product being built. That's what it means to have a cross-functional team.
>
> Scrum Teams are also self-managing, meaning they internally decide who does what, when, and how.
>
> — 2020 Scrum Guide page 5

Scrum self-describes as an 'incomplete framework.' That means it doesn't have a lot of rules or requirements.

Scrum practitioners are encouraged to 'find their path' and do what works for them. The developers should be allowed to manage themselves as much as possible.

*Figure 10. A cross-functional Scrum Team has all the skills needed to complete a Product Backlog item from beginning to end. (Credit: Pierre Selim)*

### Test Yourself

Your company just hired 50 new developers. How should they be put into teams? (Choose 1)

- ▢ A) Have the Project Manager split them up to create balanced teams based on education, experience, and skill level.
- ▢ B) Have the Scrum Master split them up into evenly divided teams where their skills match their problem domain.
- ▢ C) Have Human Resources split them up into equally divided teams based on their personality types.
- ▢ D) Let the developers split themselves up into any number of groups in any way they like.

In this case, option D would be the correct answer. If the teams are truly self-managed, they should be allowed to group in whichever way they see fit. Scrum does limit the size of a team to 10 individuals, so that is the only hard constraint that would be put on the grouping.

## Test Yourself

You, the Scrum Master, come to work on the first day of the new Sprint, and you find out from the Product Owner that the 8-member development team has split up into one group of 3, and another group of 5. The Product Owner is worried about how this will impact the project.

What one option would you, as the Scrum Master, choose?

- ▢ A) Coach the developers on the importance of staying together as a single, cohesive team.
- ▢ B) Explain that groups can't be broken up midway through the development of a product.
- ▢ C) Have Human Resources talk to the developers about making rash, surprise decisions
- ▢ D) Allow the team to split up in whatever way they see fit, and coach the Product Owner on the importance of self-managed teams.

Option D is correct.

In this case, the development team wants to split up into two groups. If that's what they want, then let them do it.

The development team is a group of self-motivated, highly motivated individuals with one overarching objective - to achieve the Product Goal. If the developers believe two teams would be more productive, then that's their decision. They get to self-organize in any way they see fit.

---

## Test Yourself

Here's another self-management question that often catches test-takers off guard.

Who has the right to remove a Developer from a Scrum Team? (Choose 1)

- ▢ A) Human Resources
- ▢ B) The Product Owner
- ▢ C) The Scrum Master
- ▢ D) The Development Team

Option D is correct.

Again, the development team is self-managing. If there is a Developer that they believe is best removed from the team, then the Developer should be removed from the team.

That doesn't mean the Developer should be fired. Maybe they were under-utilized. Or maybe they somehow impeded progress. But if the development team believes that removing an individual from the team would better serve the Product Goal, then that's a decision the team should be allowed to make.

## Why only 10 to a Scrum team?

The maximum size of a Scrum Team, including the Scrum Master and Product Owner, is 10. Here's why:

> The Scrum Team is small enough to remain nimble and large enough to complete significant work within a Sprint, typically 10 or fewer people.
>
> In general, we have found that smaller teams communicate better and are more productive.
>
> — 2020 Scrum Guide page 5

You'll be tested on the maximum size of a Scrum Team: it's 10 or fewer. That includes the Product Owner and the Scrum Master, so generally speaking, that implies 8 developers.

However, there are many instances where a Scrum Master or even a Product Owner might be doing development too. This means there might be 10 developers, with two of them playing double-duty as a Scrum Master or Product Owner.

## Can Developers also be Scrum Masters?

People often wonder if a Scrum Master can also be a Developer, or if a Scrum Master can also be the Product Owner. There's nothing in the Scrum Guide that forbids it, so it's fair dinkum.

- Is it ideal for the Scrum Master to also be the Product Owner? Probably not.
- Is it pragmatic for the Scrum Master to also be the Product Owner? Maybe sometimes it is?

If a Product Owner has amazing skills with regards to the development of a Product Backlog item or two, why not allow them to contribute to the Increment? Again, there's nothing in the Scrum Guide that forbids a Product Owner or Scrum Master from doing working on Product Backlog items.

It's nice to think of the Scrum Master and Product Owner as dedicated roles. But on small startups, where there may only be five people in the entire company, you may have all of them doing some sort of development during a Sprint. In that case, the person who took on the Scrum Master accountability also takes on accountabilities as a Developer.

There's absolutely nothing in the Scrum Guide that says a Scrum Master can't also do development, and pragmatically speaking, for many small companies, it's a reality. But regardless of how many people are wearing multiple hats, the total number of people on a Scrum Team should be 10 or less.

*Figure 11. One person may wear different hats and assume multiple roles on a Scrum Team.*

### Test Yourself

What is the optimal size of a Scrum team? (Choose 1)

- ▢ A) 3-9 people
- ▢ B) 7 plus or minus 2
- ▢ C) 10 people or less
- ▢ D) Scrum lets self-managed teams determine their own size

Option C is correct.

Normally, Scrum allows self-managed teams to decide what is best for them. But when it comes to team size, Scrum recommends teams are made up of 10 people or less. I believe SAFe uses 7 plus or minus 2 as the team-size standard, and the old Scrum Guide said 3-9, but this isn't a SAFe certification guide, and you're not being tested on what the old Scrum Guide said.

10 people or less is the correct size of a Scrum Team on the 2020 Scrum Certification exam.

### Test Yourself

The Scrum team is kept small because: (Choose 2)

- ▢ A) Small teams are easier for the Scrum Master to control
- ▢ B) Small teams communicate better
- ▢ C) Small teams are more productive
- ▢ D) Small teams demand fewer employee benefits

Options B and C are correct. According to the Scrum Guide, small teams communicate better and are more productive.

---

## Multiple Scrum Teams Working on One Product

Ever wonder how Scrum manages a big project with multiple development teams working on it?

The Scrum Guide doesn't say much about the issue, but it does outline a few important rules to follow when more than one Scrum Team works on the development of the same product.

> If Scrum Teams become too large, they should consider reorganizing into multiple cohesive Scrum Teams, each focused on the same product. Therefore, they should share the same Product Goal, Product Backlog, and Product Owner.
>
> — 2020 Scrum Guide page 5

You can expect 5 or 6 questions about this one paragraph on the Scrum Master certification exam.

It’s significant.

The first takeaway from this paragraph is that you can have multiple Scrum Teams working on the same product.

Scrum advises you to break up your teams if they become too big, and that doesn’t even mean if they are over 10 people.

Maybe 9 people on the Scrum Team is too large to be nimble and productive? In that case, if the self-managed team thinks splitting into two groups of 4 and 5 developers makes sense, then they should be encouraged to do it.

Also notice that when multiple teams work on the same product, they share the same Product Goal, Product Backlog, and Product Owner. Commit those facts to memory, because you are guaranteed to see a question like this on the exam:

### Test Yourself

Five Scrum Teams are working on the same product. Which of the following statements is not true?
(Choose 3)

- ▢ A) Each team must share the same Product Owner
- ▢ B) Each team must share the same Scrum Master
- ▢ C) Each team must share the same Product Goal
- ▢ D) Each team must share the same Product Backlog

Options A, C, and D are true, which makes option B the correct answer.

When multiple teams work on the same product, they share the same Product Owner, Product Goal, and the same Product Backlog.

---

## To Share or Not to Share a Scrum Master?

Notice how the rule about multiple teams sharing the same Product Owner does not apply to the Scrum Master when multiple teams work on the same product.

Multiple teams on the same project might share the same Scrum Master, or each team might have its own Scrum Master. The Scrum Guide doesn’t care either way.

### Test Yourself

- True or False: A Product Owner must dedicate 100% of their time to a single Scrum Team.
- True or False: A Scrum Master must dedicate 100% of their time to a single Scrum Team.

Both of those statements are false.

We know the first statement is false because the Scrum Guide insists that if five teams are working on a single product, the Product Owner must be split across all five teams, which makes it impossible to be dedicated 100% to just one.

Furthermore, there is no rule in the Scrum Guide that says a Scrum Master must be 100% dedicated to a single Scrum team. It's quite common for one Scrum Master to work on multiple Scrum Teams within an organization.

---

## Scrum Team Responsibilities

You will often see questions on the Scrum Master certification exam about who should be responsible for maintenance or research or user acceptance testing (UAT) or quality assurance (QA).

An 'incorrect answer' will suggest a subteam or external resource should do UAT or QA. That's wrong.

If QA or UAT is part of a Product Backlog item or it's part of the Definition of Done, then the Scrum Team is responsible for it.

> The Scrum Team is responsible for all product-related activities including:
>
> - stakeholder collaboration,
> - verification,
> - maintenance,
> - operation,
> - experimentation,
> - research and development,
> - and anything else that might be required.
>
> — 2020 Scrum Guide page 5

This statement reinforces the idea that a Scrum Team is cross-functional.

Every skill required to build your product, at least within the scope of your Scrum project, must exist on your team, or at least, your team should be able to acquire those skills.

There are no separate research or development or quality assurance teams in Scrum. The Scrum team is responsible for all of the work on its plate.

| | |
|---|---|
| **TIP** | Daily reports, status meetings, and allowing managers to attend the Daily Scrum are all practices that run counter to the principles of Scrum. If you ever see options on the exam that suggest such practices, immediately discount them because they are wrong. |

### Test Yourself

In Scrum, who is responsible for quality control and testing before an Increment is released into production? (Choose 1)

- ▢ A) The Quality Assurance Team
- ▢ B) And external user acceptance testing (UAT) team
- ▢ C) The DevOps team
- ▢ D) The Scrum Developers

Option D is correct.

There are no external teams in Scrum. All of the skills required to incrementally build the product must exist amongst the Scrum Team's developers.

---

## Sustainable, Self-Managed Teams

> They are structured and empowered by the organization to manage their work. Working in Sprints at a sustainable pace improves the Scrum Team's focus and consistency.
>
> — 2020 Scrum Guide page 5

This section is interesting because it speaks to the responsibilities an organization has to the Scrum Team:

- The organization allows the Scrum Team to manage their work
- The Scrum team is allowed to work at a sustainable pace

These points will arise on the Scrum Master certification exam in questions like these:

## Test Yourself

How does the organization help to manage the work of the development team? (Choose 1)

- ▢ A) The organization reads daily reports created by the Scrum Master
- ▢ B) The organization regularly sends a manager to attend the Daily Scrum
- ▢ C) The organization pulls developers into a weekly meeting to hear status reports
- ▢ D) The organization allows Scrum teams to manage themselves

Option D is correct.

The best thing an organization can do to help manage a Scrum Team is to not manage a Scrum Team. The other three options are all Scrum anti-patterns.

### Test Yourself

When should the Scrum Team add a new Developer to the project? (Choose 1)

- ▢ A) When the Project Manager provides enough funding for a new Developer
- ▢ B) When the development team repeatedly fails to meet the Sprint Goal
- ▢ C) When the development team fails to estimate the amount of work they can complete in a Sprint
- ▢ D) When a new Developer is required to maintain a sustainable pace of development

Option D is correct.

Motivated developers will work hard to achieve the Sprint Goal and the Product Goal. If developers are getting overworked and are no longer working at a sustainable pace, it is the organization's responsibility to hire new developers to help the team.

> Working in Sprints at a *sustainable pace* improves the Scrum Team's focus and consistency.
>
> — 2020 Scrum Guide page 5

---

## Increments Must Be Valuable and Useful

In every Sprint, the Scrum Team must create an Increment that is valuable and useful. Well, they must try to. That is always the goal.

The implication of needing to create a valuable Increment during every single sprint is that you can't have an 'infrastructure sprint' that just gets things organized. There is no such thing as a 'Sprint Zero' in Scrum.

Even if the increment is small and minor, each Sprint should try to create something that is valuable and will be used in the final product.

> The entire Scrum Team is accountable for creating a valuable, useful Increment every Sprint.
>
> — 2020 Scrum Guide page 5

Every Sprint must create at least one valuable, useful Increment.

An Increment must be something 'useful.' The means simply setting up infrastructure doesn't count as a 'useful' Increment. The Increment must be useful in terms of the usability and integration with the end product.

Furthermore, the Scrum Guide says the entire team is responsible for the creation of at least one, valuable, useful Increment each Sprint - not the Scrum Master, not the Product Owner, and not the developers alone, but the whole Scrum Team is responsible for the creation of a useful increment.

### Test Yourself

You're guaranteed to get a question like this on the exam:

Who is accountable for the creation of a useful Increment each Sprint? (Choose 1)

- ▢ A) The stakeholders
- ▢ B) The Product Owner
- ▢ C) The Scrum Master
- ▢ D) The Development Team
- ▢ E) The Scrum Team as a whole

In this case, E is the correct option.

The Scrum Team as a whole is responsible for the creation of a valuable and useful Increment.

---

## Scrum Accountabilities

Scrum plays a few word games with the 'accountabilities' it defines.

The Scrum Guide never mentions the term 'role' or 'job.' When it comes to the Developers, Product Owners, and Scrum Masters, the Scrum Guide is very careful to use the much more abstract term 'accountability.'

The term 'role' is not found in the 2020 Scrum Guide even once.

> Scrum defines three specific accountabilities within the Scrum Team: the Developers, the Product Owner, and the Scrum Master.
>
> — 2020 Scrum Guide page 5

### Test Yourself

Be as exact as possible when you answer this question. The Scrum Guide defines: (Choose 1)

- ▢ A) 3 Roles
- ▢ B) 4 Roles
- ▢ C) 3 Accountabilities
- ▢ D) 4 Accountabilities

The answer to this question is C.

Scrum defines 3 accountabilities, not roles.