SCRUM MASTER
Certification Guide

Darcy DeClute

Edited by:
Prachi Shah
Amanda Punsammy
Forward by Sal Pece, Xennial CEO

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 have, 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

This is another politically charged couple of lines that need unpacking.

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 that 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.' But they are technically not 'roles.'
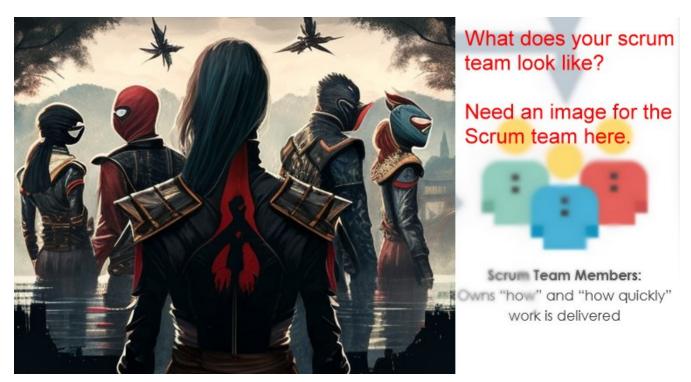

Figure 1. A Scrum Teams has one Product Owner, one Scrum Master and developers.

Be @Scrumtuous.

Test Yourself

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

> What is the typical composition of a Scrum Team?
>
> ☐ A Scrum team is made up of one Scrum Master, many Product Owners, and many Developers
>
> ☐ A Scrum team is made up of one Scrum Master, one Product Owner, and many Developers
>
> ☐ A Scrum team is made up of many Scrum Masters, many Project Managers, and many Developers
>
> ☐ A Scrum team is made up of one Scrum Master, one Project Manager, one Product Owner, and many Developers

The answer is C.

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 of the following options best describes how a team of software developers should operate under the Scrum framework?
>
> ☐ Software developers should do as much development as they can and then pass quality control work off to QA people.
>
> ☐ Software developers should only build core features and then allow other to take care of non-functional aspects.
>
> ☐ Software developers should outsource work they are not skilled to perform to third parties or external teams.
>
> ☐ All work will should completed by developers on a Scrum Team, always ensuring the Definition of Done is met.

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

Many Scrum 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, one Product Owner, and developers.

Sub-teams 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 sub-teams 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 sub-teams 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 ther are no sub-teams and there are no hierarchies in Scrum.

What does it mean to say there are no sub-teams 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 'Senior Developer.' Everyone one the team is equal.

Test Yourself

Here's the structure of a prototypical Scrum Master Certification exam question that tests you on this hierarcies and sub-teams:

> There is an issue. What should you do?
>
> ☐ Talk to the senior developer on the Scrum Team about it
>
> ☐ Talk to someone on the QA team about it
>
> ☐ Talk to someone on the testing team about it
>
> ☐ Coach the developers around strategies of how to fix it

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 2. The only time you have a sub-team in Scrum is if there is literally a team of developers working on a sub.

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?

☐ Tell the dev team the Mars Orbiter will be removed from the required features list.

☐ Cancel the Sprint until the team finds someone who can build a Mars orbiter.

☐ Tell the dev team the Mars Orbiter feature will be outsourced to Elon Musk and SpaceX.

☐ Coach the dev team on the importance of learning and acquiring the skills required to complete all of the required product features.

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

Here's the official, Scrum Guide definition of what it means to be a cross-functional team:

> 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 3. A cross-functional team has all the skills needed to complete a Product Backlog item from beginning to end.

Test Yourself

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

☐ Have the Project Manager split them up to create balanced teams based on education, experience, and skill level.

☐ Have the Scrum Master split them up into evenly divided teams where their skills match their problem domain.

☐ Have Human Resources split them up into equally divided teams based on their personality types.

☐ 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 would you, as the Scrum Master, do?

☐ Coach the development team on the importance of staying together as a single, cohesive team.

☐ Explain that groups can't be broken up midway through the development of a product.

☐ Have Human Resources talk to the developers about making rash, surprise decisions

☐ Allow the team to split up in whatever way they see fit, and coach the Product Owner on the importance of self-managed teams.

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?

☐ Human Resources

☐ The Product Owner

☐ The Scrum Master

☐ The Development Team

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 the Scrum Master gets sick, or 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 a 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.

Test Yourself

> What is the optimal size of a Scrum team?
>
> ☐ 3-9 people
>
> ☐ 7 plus or minus 2
>
> ☐ 10 people or less
>
> ☐ Scrum lets self-managed teams determine their own size

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)
>
> ☐ Small teams are easier for the Scrum Master to control
>
> ☐ Small teams communicate better
>
> ☐ Small teams are more productive
>
> ☐ Small teams demand fewer employee benefits

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 Certification exam. It's significant.

Be @Scrumtuous.

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?

☐ Each team must share the same Product Owner

☐ Each team must share the same Scrum Master

☐ Each team must share the same Product Goal

☐ 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 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 sub-team 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.

Be @Scrumtuous.

Test Yourself

Take this question for example:

> In Scrum, who is responsible for quality control and testing before an Increment is released into production?
>
> ☐ The Quality Assurance Team
>
> ☐ And external user acceptance testing (UAT) team
>
> ☐ The Product owner
>
> ☐ The Development team

There are no external teams when it comes to Scrum and the product your team is building. All of the skills required to incrementally build the product the Scrum Team is working towards 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?
>
> ☐ The organization reads daily reports created by the Scrum Master
>
> ☐ The organization regularly sends a manager to attend the Daily Scrum
>
> ☐ The organization pulls developers into a weekly meeting to hear status reports
>
> ☐ The organization allows Scrum teams to manage themselves

The last answer is correct. The best thing an organzation can do to help manage a Scrum Team is to not manage a Scrum Team.

By the way, the other three options are all Scrum anti-patterns. Daily reports, status meetings, and

sending managers to the Daily Scrum are practices that should not happen.

Test Yourself

> When should the Scrum Team add a new developer to the project?
>
> ☐ When the Project Manager provides enough funding for a new developer
>
> ☐ When the development team repeatedly fails to meet the Sprint Goal
>
> ☐ When the development team fails to estimate the amount of work they can complete in a Sprint
>
> ☐ When a new developer is required to maintain a sustainable pace of development

Motivated developers will work hard to achieve the Sprint Goal and the Product Goal. But they should not be overworked.

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.

Increments Must Be Valuable and Useful

Every Sprint the Scrum Team must create an increment that is valuable and useful.

That means you can't have an 'infrastructure sprint' that just gets things organized.

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

There are a couple of very important concepts nested in this sentence.

First, every Sprint must create at least one valuable, useful Increment. Notice that 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 of the end product.

Secondly, notice how the 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 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 eash Sprint?

☐ The stakeholders

☐ The Product Owner

☐ The Scrum Master

☐ The Development Team

☐ The Scrum Team as a whole

In this case, it's the last option: the Scrum Team as a whole.

Scrum Accountabilities

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

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

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

This sentence reiterates the opening paragraph of the section on the Scrum Team. Again, note that Scrum describes the developers, Scrum Master and Product Owner as accountabilities, not roles. The term 'role' is never mentioned in the Scrum Guide.

Test Yourself

The Scrum Guide defines:

☐ 3 Roles

☐ 4 Roles

☐ 3 Accountabilities

☐ 4 Accountabilities

The answer to this question is C. Scrum defines 3 accountabilities, not roles.