

# Darcy DeClute's Scrum Master CERTIFICATION GUIDE

The Definitive Resource



@scrumtuous



Editors  
Prachi Shah  
Walker Aldridge  
Amanda Punسامی

foreword by **Xennial CEO Sal Pece**



# Scrum Master Certification Guide



# Table of Contents

A note from the author .....	1
So many people to thank... .....	1
I'm @Scrumtuous everywhere! .....	2
Foreword .....	3
The role of the Scrum Master .....	3
Measuring Agile Transformation Success .....	3
Keeping Scrum Working .....	4
Chapter 0: Getting Started .....	7
Forget Everything You Know .....	8
So what is Scrum? .....	8
Do a little planning. Do a lot of building. ....	9
Incremental Progress Over Time .....	9
Review and Reflect .....	10
The Scrum Master .....	10
Get The Scrum Guide .....	11
Read the Scrum Guide .....	11
Chapter 1: What is Scrum? .....	13
The Definition of Scrum .....	14
Scrum is a Framework .....	15
Scrum and the Scrum Master .....	16
Don't Overstate the Role of the Scrum Master .....	17
Scrum is Simple, Pragmatic and Lean .....	17
It's a Guide. It's not an Instruction Manual .....	18
Exposing Efficacy .....	19
Combine the Scrum Framework with other Processes .....	19
Chapter 2: Scrum Theory .....	21
Rationalists vs Empiricists .....	21
Empirical Example .....	21
Lean Thinking .....	22
Empiricism and Lean Thinking .....	23
Predictability and Risk .....	24
The Incremental Nature of Scrum .....	25
Cross-Functional Teams .....	25
Moving from Sprint to Sprint .....	28
Forget Everything You Think You Know .....	30
Keep the Focus on the Goal .....	31

Chapter 3: Scrum Pillars and Values .....	33
The Scrum Pillars .....	33
Scrum Values .....	34
The Endless Feedback Loop .....	34
Inspection .....	35
Transparency .....	36
Adaptation .....	37
Commitment, Focus, Openness, Respect and Courage .....	39
Scrum Teams .....	41
Composition of a Scrum Team .....	41
There are no Subteams in Scrum .....	44
Cross-functional Scrum Teams .....	45
Cross-Functional and Self-managed .....	46
Why only 10 to a Scrum team? .....	49
Can Developers also be Scrum Masters? .....	49
Multiple Scrum Teams Working on One Product .....	51
To Share or Not to Share a Scrum Master? .....	52
Scrum Team Responsibilities .....	52
End-to-End Feature Development .....	53
Sustainable, Self-Managed Teams .....	53
Test Yourself .....	54
Increments Must Be Valuable and Useful .....	55
Linguistic Gymnastics .....	56
The Developers .....	59
Who is a Developer? .....	59
Feature Development vs. Overall Accountability .....	60
The Developer's Job is Simple: Create .....	60
Developer Accountabilities .....	61
Instilling Quality .....	61
Daily Adaptation .....	62
Interpersonal Accountability .....	63
What's not said about Developers .....	63
Adding New Developers .....	63
The Impact of Adding Developers .....	64
The Product Owner .....	67
Product Owners Maximize Value .....	67
How do Product Owners Maximize Value? .....	68
Product Backlog Management .....	68

Product Owner Accountabilities .....	68
Who Does the Product Owner’s Work? .....	70
Product Ownership and Trust .....	70
A Single Product Owner .....	71
Keeping Tabs on the Product Owner .....	71
Trust, Transparency, and the Product Backlog .....	73
Inspection of the Increment .....	73
Negotiating Product Features .....	73
Chapter 7: The Scrum Master .....	75
Scrum Requires a Scrum Master .....	76
What the Scrum Master Doesn’t Do .....	77
How Scrum Masters Serve the Scrum Team .....	78
Examples of Impediments a Scrum Master Can Resolve .....	81
Positive, Productive, and Timeboxed .....	82
Scrum Master and the Product Owner .....	83
Removing Barriers Between Stakeholders and Teams .....	84
Planning and Advising .....	85
Chapter 8: Scrum Events .....	87
The Sprint Container .....	87
The Importance of the Scrum Events .....	88
Regularity and consistency .....	89
Every Scrum Event Serves a Purpose .....	90
Same Time, Same Place .....	91
Chapter 9: The Sprint .....	93
Fixed Length Sprints .....	93
One Month or Less .....	93
The Start of the Sprint .....	94
What Happens During the Sprint? .....	95
Why do we have Short Sprints in Scrum? .....	97
The Benefits of Short Sprints .....	97
Empiricism Trumps Scrum Metrics .....	97
Scrum Can’t Predict the Future .....	99
The Importance of Empiricism .....	99
Who Can Cancel a Sprint? .....	99
What isn’t said about the Sprint .....	100
When to Start the Sprint? .....	100
No Rule Means There’s No Rule .....	101
Non-Empirical Scrum Metrics .....	101

Chapter 10: Sprint Planning .....	103
Product Backlog Items and the Product Goal .....	104
Who's Invited to the Sprint Planning Meeting? .....	104
Minimizing Inter-team Dependencies .....	104
Answering Why, What, and How .....	106
Timeboxing Sprint Planning to 8 Hours .....	106
Why is the Sprint valuable? .....	107
Sprint Planning and the Sprint Goal .....	109
Product Backlog Item Selection .....	110
Product Backlog Refinement .....	111
What can be Done in this Sprint? .....	112
How will the chosen work get done? .....	112
Product Backlog Decomposition .....	113
No Points or User Stories in Scrum .....	113
The Outcome of Sprint Planning .....	114
Chapter 11: The Daily Scrum .....	117
Adapting to Change .....	118
Keep the Time and Location Consistent .....	118
Demystifying the Daily Scrum .....	119
Keep the Daily Scrum Simple .....	119
Rules of the Daily Scrum .....	120
Daily Scrum Time and Location .....	120
When Scrum Masters and POs do Development .....	122
The Blended Role Anti-Pattern .....	123
Who Participates in the Daily Scrum? .....	123
Purpose of the Daily Scrum .....	124
Does Scrum Really Have Too Many Meetings? .....	125
Chapter 12: Sprint Review .....	127
Sprint Review Participants .....	128
What to show in the Review? .....	128
Only Completed Increments in the Sprint Review .....	129
A Discussion on What to do Next .....	129
The Sprint Review is not Simply a Presentation .....	131
Chapter 13: Sprint Retrospective .....	133
Updating the Sprint Length and the DoD .....	134
The 80/20 Principle .....	135
Adding to the Next Sprint's Backlog? .....	135
The Conclusion of the Sprint .....	137

After the Sprint .....	137
Chapter 14: The Scrum Artifacts .....	141
Answering Three Important Questions .....	141
Scrum Commitments .....	142
Why do we need Scrum commitments? .....	143
Scrum Commitments, Goals and Plans .....	143
Empiricism and the Scrum Commitments .....	144
Chapter 15: The Product Backlog .....	147
An Ordered List of Features .....	147
Single Source of Work .....	148
Product Backlog Refinement .....	148
Estimation and Sizing .....	150
Commitment: The Product Goal .....	151
Product Goal and the Product Backlog .....	152
Value Delivery .....	153
A Single, Shared Objective .....	154
Chapter 16: The Sprint Backlog .....	157
3 Questions to Answer .....	157
Who Owns the Sprint Backlog? .....	158
What gets Updated Throughout the Sprint .....	158
The Sprint Goal Never Changes After Planning .....	159
Transparency, Inspection, and Adaptation .....	159
Sprint Goal .....	160
Product Owner and Developer Collaboration .....	161
Chapter 17: The Increment .....	163
Iterative and Incremental .....	163
Integrated Increments .....	164
Understanding Integration .....	164
A Poorly Integrated Increment .....	164
When to Integrate .....	165
The Right Time to Integrate .....	166
Multiple Increments .....	167
Continuous Delivery of Valuable Increments .....	167
Commitment: The Definition of Done .....	167
Scrum Pillars, Transparency, and 'Done' .....	169
Organizational Standards .....	170
Done Over Time .....	170
Bad Definitions of Done .....	170

Multiple Teams and the Definition of Done .....	171
What Scrum Doesn't Say About Multiple Teams .....	171
PART II: THE EXAM QUESTIONS .....	173
The Secret Test-Taking Trick .....	173
Ten Tough Scrum Master Practice Exam Questions .....	175
Answers to the First Practice Exam .....	179
A Simple Trick .....	187
A Hack to Help You Pass the Exam .....	187
The Party Boat Disaster .....	189
Questions .....	189
Answers .....	195
25 Question Exam #1 .....	203
Good Luck! .....	203
Answers .....	214
25 Question Exam #2 .....	233
Good Luck! .....	233
Answers .....	243
Final Exam: 40 Questions .....	261
Good Luck! .....	261
Answers .....	276
Put your fears aside .....	306
Some unorthodox advice .....	306
Sign up and do a dry run .....	307
Getting into their head .....	307
An empirical evaluation .....	307
Zero risk. Massive reward. ....	308
You may surprise yourself....	309
You might be over-preparing? .....	309
You've got this! .....	309
I want to share in your success .....	309
A few people I need to thank....	311
More to come! .....	313

## A note from the author

Hi! I'm Darcy DeClute.

As I write this note I'm filled with a mix of conflicting and confusing emotions that range from humility to pride to embarrassment.

It humbles me to know you're reading this book.

If you're reading a book about certification, you're obviously interested in acquiring a Scrum Master accreditation, which means you're a highly-motivated professional intent on learning, resume building and career advancement.

And you've put your faith in *me* to guide you on a learning journey. That's humbling.

As overwhelming as it is to know you've put your faith in me, I'm equally proud of the book I've written here.

I think this book delivers on what it promises. I think this book will play a key role in getting you past the post when it comes time to sit for the Scrum Master certification exam.

A number of students have read this book and then passed the Scrum Master exam, with a few even garnering a perfect score. I know this book played a key role in their success and I'm confident it will play a key role in your success as well. That makes me beam with pride.

## So many people to thank...

I'm also embarrassed when I see this book with my name on the cover, because I feel like an imposter taking the title credit.

When I mentioned to some of my peers and mentors that I wanted to write a Scrum Master certification guide to fill what I thought was a void in the market, my friends encouraged and inspired me.

Some people I follow on Twitter, people like Walker Boh, graciously donated their own free time to proofread, edit, criticize and add corrections. Other peers and mentors added paragraphs and extended the text when they thought a chapter was light or a topic needed clarification.

Some friends helped to generate images with MidJourney prompts and fixed issues with formatting.

So when I see my name on the front cover of this book, I feel as though I'm stealing credit, because this was completely a team effort.



*Figure 1. At home on native land. Baffin Island, Canada*

Be @Scrumtuous and Follow @Xennial on Twitter

I can't thank the people who helped me get this book past the finish line enough. I love all of you.

## I'm @Scrumtuous everywhere!

This book is self-published.

There's an old saying that asserts the biggest problem with self-published books is the person who publishes them. Having read some really low-quality, self-published tripe, I don't completely disagree with that sentiment.

However, since I do self-publish, I have the benefit of being able to make updates and changes at a moments notice.

If you do find something in this book that is unclear, misspellled, confusing or even incorrect, please find me on Twitter or LinkedIn and let me know.

I'm @Scrumtuous everywhere.

Or better yet, just email me:

[darcydeclute@gmail.com](mailto:darcydeclute@gmail.com)

As I mentioned before, people have put a lot of faith in me to help them get certified and drive their career forward. I want to do everything I can to make that possible. Any help you can provide would be appreciated not only by me, but by everyone who reads this book in the future.

Oh, and one last thing. If you do enjoy the book, please leave a 5-star review on Amazon. It would not only put a giant smile on my face, but it would also help to get this book in front of other people who need it.

Thanks again for reading this book. I really hope you enjoy it.

Love Darcy.



*Figure 2. Please come find me on Twitter. I'm @Scrumtuous on all platforms!*

## Foreword

I felt a little nervous when Darcy asked me to write this foreword.

You see, I'm not the biggest Scrum Master cheerleader in the world.

In fact, in many of the Agile transformations I've led, one of our success metrics is how close we come to making the Scrum Master completely unnecessary.

That's not to say I'm against Scrum. I believe Scrum is the right Agile framework to use when starting a new project.

### The role of the Scrum Master

I advocate for Scrum, but I also advocate for teams to evolve to such an advanced level of Agile understanding that they don't really need a Scrum Master.

The role of the Scrum Master is to teach, coach, and mentor others in the ways of Scrum, while acting as a leader who serves as the team goes through an Agile transition.

But here's the thing about transitions: they eventually reach completion.

When an organization truly achieves Agility, the need for a Scrum Master to coach them on cross-functional teams, self-management, and the importance of reviews and retrospectives progressively diminishes. These Agile practices eventually become second nature.

Successful Agile teams don't need coaching on these practices because they naturally incorporate them into their work. It becomes odd for them not to.



*Figure 3. Sal Pece, Xennial CEO*

### Measuring Agile Transformation Success

That's when you know your Agile transformation is a success. That's when you know the team has approached Agile enlightenment.

Of course, you can't do Scrum without a Scrum Master. Scrum without a Scrum Master isn't Scrum.

However, over time, a team's dependence on the Scrum Master should diminish.

At the beginning of an Agile transformation, each development team might require its own dedicated Scrum Master.

Well-coached teams will rely less and less on the Scrum Master for leadership. As Agile transformations succeed, a Scrum Master will divide their time across multiple teams because

individual teams become less dependent on their guidance.

Once enlightenment is achieved, a single, underutilized Scrum Master will be shared across multiple teams throughout the organization.

There's a recurring meme in Agile circles that jokes "never ask a Scrum Master what they do, because you'll find out that they don't do much."

I love that meme because it's not only funny but because it should be true if your organization has implemented Scrum correctly. That meme is basically my goal during an Agile transformation.

## Keeping Scrum Working

On teams where developers and the Product Owner truly understand how Scrum works, the role of the Scrum Master becomes almost ceremonial. If your teams are building amazing products while the Scrum Master is searching for things to do, you're probably implementing Scrum exceptionally well.

And that's why I was nervous about reviewing this book and contributing a foreword.

I was concerned that this Scrum Master Certification Guide would, like most other books about Scrum, overstate the role of the Scrum Master.

But my nervousness quickly turned into enthusiasm as I turned page after page of this book.

As I read through the text, I felt Darcy was reflecting my thoughts about effective Scrum directly back to me.

Miss DeClute strikes just the right balance between recognizing the importance of the Scrum Master's role within an organization and emphasizing that with highly motivated, self-managed teams, reliance on the Scrum Master accountability should gradually diminish over time.

Darcy gets it right.

Enjoy this certification guide.

It will not only help you pass the Scrum Master certification exam but it will also provide you with a strong understanding of how Scrum *should* work once an enterprise successfully completes an Agile transformation.

Sal Pece

CEO, Xennial Innovations Inc.

---

When the student is ready, the teacher will appear.

*Tao Te Ching*

---



## 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 exam.



**The Scrum Diaries, by Darcy DeClute (@scrumtuous)**

*Figure 4. Download and read the official Scrum Guide a few times before reading this book. Read it a few times after reading this book too. The Scrum Guide is the sole source of truth for any Scrum Certification exam.*

## Forget Everything You Know

If you've ever worked with Scrum in a professional setting and you want to ace the Scrum Master certification exam, the first thing you need to do is forget everything you think you know about Scrum.

- Forget about story points.
- Forget about use cases.
- Forget about poker planning.

These concepts have nothing to do with Scrum. They are never mentioned in the Scrum Guide, and references to them on the certification exam are only there to distract you and trick unprepared test-takers into selecting incorrect answers.

On the Professional Scrum Master certification exam, you'll be tested almost exclusively on concepts within the Scrum Guide. If it's not in the Scrum Guide, it's not a correct answer on the exam.

The definitive source of truth regarding Scrum comes exclusively from the Scrum Guide. That's what we'll cover in-depth during this guide.

But before we get into the nuanced specifics, I wanted to provide my own, unofficial take on what Scrum is. This is how I describe Scrum to people who ask me about it. I think it's a good overview. It lays the groundwork for future, more technical discussions.

## So what is Scrum?

Scrum is a lightweight, purposefully incomplete framework that helps development teams get started and keep going.

- Does someone have a product vision?
- Do you have a team of developers who are anxious to get building?
- Are stakeholders champing at the bit?

If that's the case, Scrum says you should start a development Sprint right now.

- Don't waste a lot of time planning because your plans are going to change.
- Don't waste a lot of time designing because your initial designs might not work.

Instead, start developing. Start a development Sprint now.

A development Sprint is a short period of time in which developers focus on building a few important features that will become part of the final product.

A development Sprint can never last more than a month. Make them shorter if you want to minimize risk.

## Do a little planning. Do a lot of building.

As I said before, teams shouldn't waste a lot of time planning. However, Scrum doesn't say teams shouldn't plan at all. The very first thing a Scrum Team does when their development Sprint starts is plan.

Not too much time should be spent planning. Never more than 8 hours, and never plan too far into the future. The further you look into the future, the wider the cone of uncertainty becomes. That's why short Sprints are often better than long ones.

During Sprint Planning, the developers talk to the Product Owner about what to build. The Product Owner is the person on the Scrum Team who has the product vision and knows what stakeholders want.

The Product Owner also has an ordered list of features that need to get developed. That's called the Product Backlog.

During planning, Scrum Developers select Product Backlog items they think they can complete before the end of the Sprint.

At the end of Sprint Planning, the developers walk away with:

- A list of features they are going to work hard to complete.
- A plan to get those features finished.
- An overarching Sprint Goal that keeps the team focused.

Then the work begins!

Every workday, the developers meet for a few minutes in something called a Daily Scrum.

- It's not a standup meeting.
- It's not a status meeting.
- The Product Owner and the Scrum Master aren't even supposed to participate.

It's just a chance for the developers to quickly catch up and get the ball rolling again, nothing more.

## Incremental Progress Over Time

Every Sprint has to create something useful.

Every Sprint must create something that will become part of the final product.

In Scrum, we call these useful, additive pieces Increments. Products get built incrementally.

At the end of the Sprint, there's a Sprint Review where the Scrum Team gives stakeholders a chance to play around with the new features.

Stakeholders are only shown features that are done - not partially done or almost done, but done.

Technically speaking, stakeholders are shown features that meet the "Definition of Done," which is a quality standard everyone on the project has agreed upon.

## Review and Reflect

Product reviews at the end of every Sprint ensure stakeholders get to see what's being developed at least once every month. It keeps everyone in the loop. Feedback loops are an essential part of Scrum.

After the Sprint Review, the Scrum Team holds a Sprint Retrospective where they discuss what worked and what they need to do better.

When the Sprint Retrospective ends, a new Sprint begins, which means another short planning meeting, Daily Scrums, and finally a Sprint Review and Retrospective.

That's why Scrum is said to be iterative. You just keep doing Sprints until the work is done.

And that's Scrum.

It's a simple framework that says:

- Plan, but don't plan too much.
- Work hard at a sustainable pace to develop something usable and valuable.
- Regularly review your work with stakeholders.
- Regularly take time to talk about team dynamics and how to improve things.

And all of this is accomplished by a team that is made up of:

- One Product Owner who defines the Product Goal and manages the Product Backlog.
- The developers who know how to build the product.
- One Scrum Master, who doesn't do much.

## The Scrum Master

As Sal Pece said in the foreword, that's sort of an ongoing joke in Scrum - that the Scrum Master doesn't do anything. But the fact is, on a really well-run team, a Scrum Master doesn't have much to do.

The Scrum Master just makes sure Scrum is applied properly.

- They don't manage the team.
- They don't manage the project.
- They don't schedule Zoom calls.

- They don't book conference rooms.
- They don't manage finances.
- They don't update JIRA tickets.

They just coach teams and organizations on how to properly apply the Scrum framework while behaving as a leader who serves the team. That's the Scrum Master's job.

Scrum describes itself as a simple, lean, incomplete framework, and that's a good description.

Scrum just describes the best practices any team should be doing if they want to work efficiently and effectively.

It may not work for everyone, but everyone should at least give it a try.

## Get The Scrum Guide

That's my quick description of Scrum, and I think it's a good one.

Having said that, the Scrum Master certification exam doesn't test you on what I think Scrum is all about, it tests you on what the Scrum Guide says Scrum is all about, which is why I want you to download and print out a couple of copies of the 2020 Scrum Guide.

The Scrum Guide is only 14 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 5. Try to hit as many learning modalities as you can whenever you learn something new.*

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. That way you'll have a better idea of what phrases like *the Sprint Goal* and *a usable Increment* mean.

Be @Scrumtuous and Follow @Xennial on Twitter

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!

## Chapter 1: What is Scrum?

How would you define Scrum in a single sentence?

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 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 6. Scrum is a lightweight, incomplete framework that helps teams solve complex problems as they work towards a Product Goal. Scrum advocates downplay the terms 'process' and 'methodology.'*

## 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 insight.

- 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, so commit to it.

## The Words Not Spoken

The Scrum Guide says a lot in its brief 14 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.

### NOTE

While the authors of the Scrum Guide both signed the Agile Manifesto, the word *Agile* never appears once in the 2020 Scrum Guide.

## 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:

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

- True
- False

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.

---

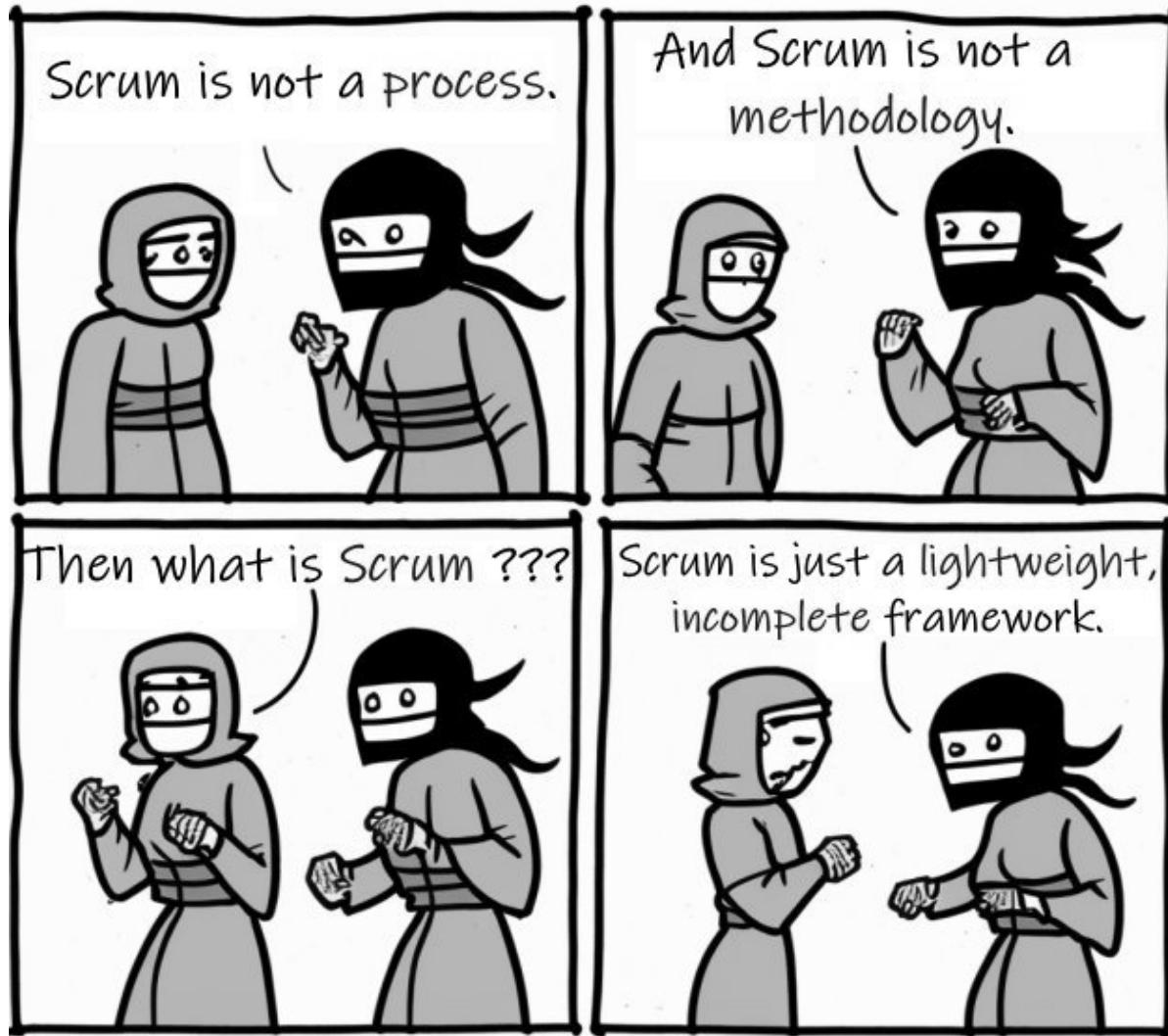
### Test Yourself

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 follow
- 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.

Scrum is a "lightweight framework that helps people, teams, and organizations generate value through adaptive solutions for complex problems." References to Scrum being a *methodology* or a *process* will always be a wrong option on the Scrum Certification exam. It's not just about software either.



## Scrum and the Scrum Master

Here's the Scrum Guide's high-level overview of how Scrum is supposed to work, along with a quick note about the role the Scrum Master plays in keeping Scrum working.

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

## Don't Overstate the Role of the Scrum Master

The name 'Scrum Master' sounds intimidating and authoritative.

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

They don't. The Scrum Master controls very little. In fact, the word 'control' is antithetical to what the Scrum Master is all about.

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

## Scrum is Simple, Pragmatic and Lean

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.

## 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 14-page Scrum Guide. Outside of the few rules Scrum does have, the framework encourages people to discover strategies that work best for them.

**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.

## Test Yourself

Scrum is a complete and proven framework that helps teams achieve goals and create value.

- True
- False

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

Scrum can be used alongside various other methodologies including Kanban and Lean.

- True
- False

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.

**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.

## Test Yourself

When implemented properly, Scrum has the capacity to expose ineffective management.

- True
- False

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

### Rationalists vs Empiricists

Rationalists say knowledge is best derived through reason, logic and deduction. Mathematics is a very *rational* pursuit.

Empiricists say evidence acquired through experience and experimentation is a better way to solve a puzzling problem.

Pragmatists sit somewhere in the middle and can't make up their mind as to which approach is better.

I actually consider myself a bit of a pragmatist, but when it comes to solving complex problems, Scrum wants nothing to do with wish-washy pragmatism. :)

Scrum is founded upon the concepts of *empiricism* and *lean thinking*.

In the context of Scrum, empiricism means understanding the situation you are in based on knowledge, experimentation, 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 always a more effective way to approach a complex problem than following a plan written up six months ago, or making decisions based on trendlines on a historical chart.

### Empirical Example

Here's a simple example of empiricism to drive the concept home for you.

Imagine you wanted to know how long it would take for a hockey puck to fall to the ground when dropped from the top of Toronto's CN Tower.

One approach would be to grab a pen and paper, look up the height of the CN Tower, grab Newton's equation for gravity, factor in wind resistance and use the power of math to determine a result.

That's a good approach, but it's very rational. It's not at all empirical.

Scrum Teams place the greatest amount of value on empirical results.

To answer this question, a Scrum Team would have their developers each channel their inner *Spiderman* and climb to the top of the CN Tower with a bag of pucks.

They'd drop the pucks from the tippy top and use a timer to see how long it takes for the black, vulcanized rubber to hit the ground.

The developers would then analyze the results of their experiment and come up with a result that was based on what they observed. Making decisions based on what you observe through your six senses is what empiricism is all about.

In Scrum, we want to base our decisions on empirical observations. That's not to say Scrum developers aren't rational or pragmatic.

In fact, taking a pragmatic approach to many question on the Scrum Master exam will help you get closer to achieving a perfect score on the test. But when given the choice to base an important decision based on rational, pragmatic or empirical analysis, Scrum wants you to prioritize empiricism every time.

## Lean Thinking

Lean thinking is also a pivotally important *concept* to master if you want to pass any Scrum exam. You don't have to be an expert on *lean manufacturing* practices, but you do have to master the art of *lean thinking*.

The idea of lean thinking comes from productivity enhancements Toyota discovered in the 1950s and 60s. Obviously Toyota was building cars, but the lessons of lean thinking are universal to all product development domains.

Key concepts of lean manufacturing that manifest themselves in the Scrum framework include:

- The idea that all unnecessary steps in a process should be eliminated
- The belief that all work should be tied tightly to a specific outcome or goal
- The insistence that products should not be over-engineered with unnecessary features
- The need for a direct connection with the customer or stakeholder
- The insistence that teams have strong leaders whose priority is to serve their team
- The desire for workers not to be left idle during the production process
- The fact that improvements should be made in accordance with empirical observations

## Empiricism and Lean Thinking

Just looking at this list of lean concepts you can see how:

- The Scrum Master role maps to the lean demand that teams have strong leaders
- The Sprint and Product Goals map to the idea that work should be tied to a specific outcome
- Scrum's Sprint Review embraces the ideas of maintaining a tight connection with the stakeholder

Avoid waste and embrace minimalist thinking when you're working on a Scrum project. That's *lean thinking*.

You are guaranteed to get five or six questions about empiricism and lean thinking 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 made up of individualized, compartmentalized 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 'Command and Control' approach that the military takes is the opposite of the collaborative, lean approach Scrum takes to decision making, 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

You just keep repeating this sequence of steps ad infinitum until you either run out of Product Backlog items to work on or the project comes to an end.

## The Incremental Nature of Scrum

Scrum is incremental.

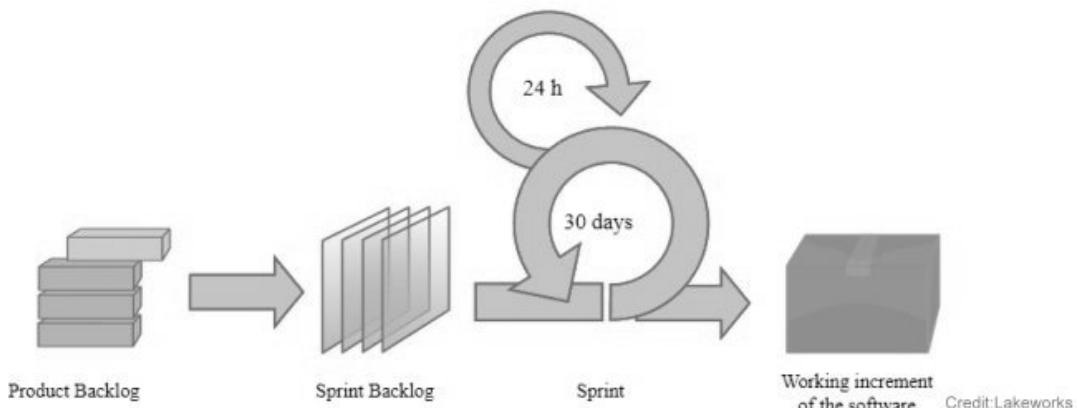
"The Scrum Team turns a selection of the work into an Increment of value during a Sprint."

— 2020 Scrum Guide page 3

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 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 7. Scrum builds products incrementally through a set of steps that are iteratively repeated throughout the product development lifecycle.(Credit Lakeworks)*

## 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 someone to document the product? That person is on the Scrum team.
- Does your project need an architect? That person is on the Scrum team.
- Does your project need people to do quality assurance(QA)? Then those QA people are developers 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.

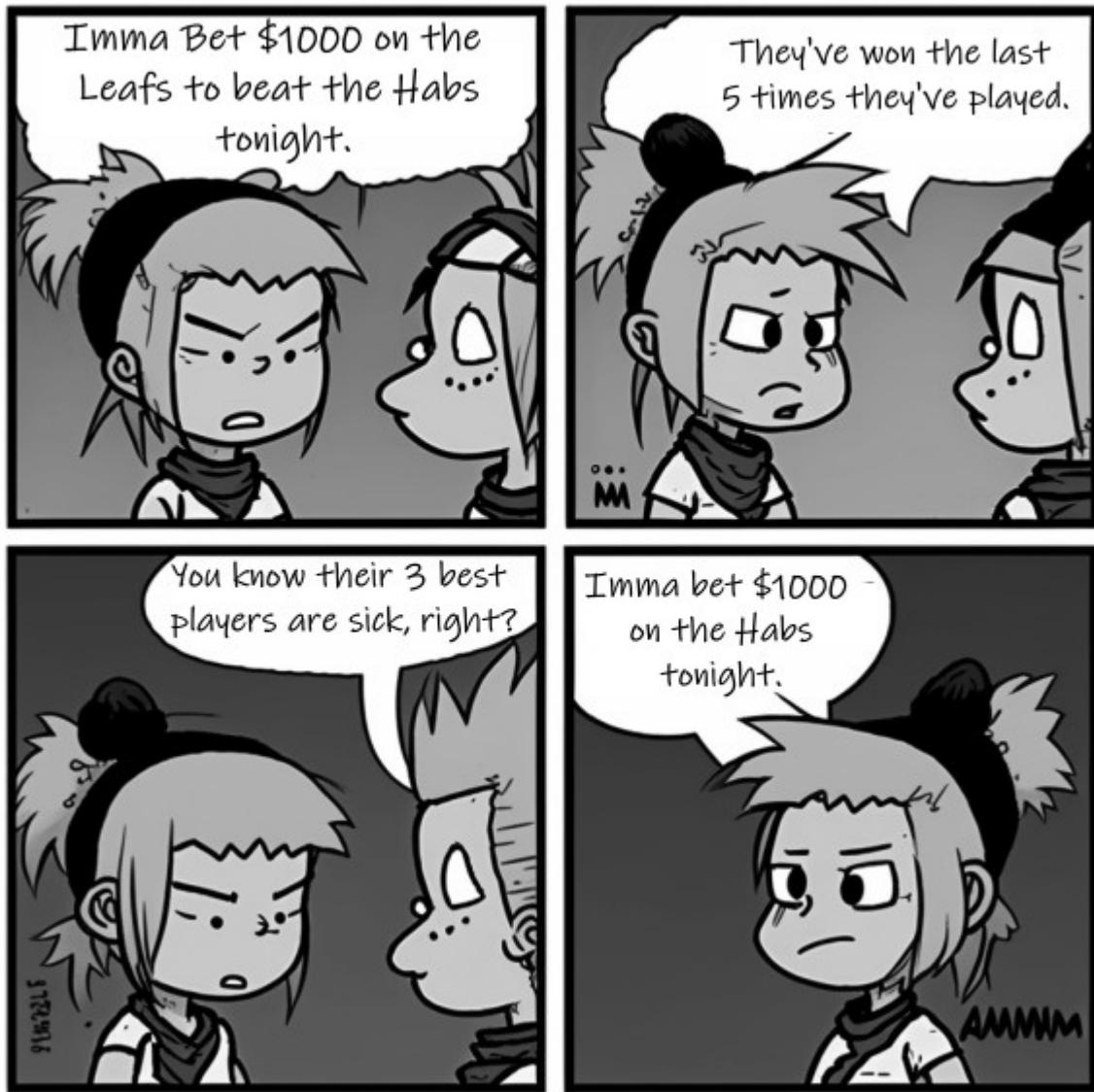


Figure 8. Predictive tools are useful, but they don't replace empirical knowledge acquired through experience and experimentation. Knowledge of pertinent facts should always take priority over charts and trendlines.

## 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 Mars spaceship
- D) Get the team to start learning about how to build a spaceship to Mars

Option D is correct, and yes, this question is silly to the extreme, but it makes an important 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 timeboxed 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.

## 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.



Figure 9. The four timeboxed events in Scrum all occur inside a fifth event known as the 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

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. A new Sprint could be said to start immediately after this event concludes, depending upon how pedantic you want to be about the answer.

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.

---

## Test Yourself

Which of the following concepts are consistent with a lean approach to product development?

(Choose 3)

- A) The elimination of unnecessary steps in a process
- B) Isolated development phases that flow into each other
- C) The removal of unnecessary features in a product
- D) Significant up-front planning and design
- E) The elimination of idle time where developers are not working
- F) Waiting for a manager to give you permission to proceed with a task
- G) Waiting for a planned event to discuss an important matter

Options A, C and E are correct.

These three options are core concepts learned from *lean manufacturing*.

Options B and D represent the waterfall model, which is the antithesis of Scrum and lean.

---

## Forget Everything You Think You Know

Part of passing the Scrum Master certification exam is to first forget everything you *think* you know about Scrum, and then focus on the core concepts contained within the Scrum Guide.

That's why you won't hear me talking about the *Agile Manifesto* or discussing *story points* in this book. Other Scrum Master certification books invest a lot of time in those topics, but I think that does you a disservice.

The Scrum Guide never once mentions the word *Agile* and it never talks about *user stories* or *story points*.

Furthermore, the Scrum Master exam never tests you on these concepts, and whenever a potential answer references a *user story* or a *story point*, those answers will be red herrings trying to trip you up and get you to select an incorrect option.

That's also why you'll often find me repeating myself ad nauseam about certain topics throughout the book. For example:

- I will annoyingly repeat myself about the time allotted to the various Scrum events.
- I constantly emphasize that you don't have to wait for a Scrum event to change your plan.
- Scrum is a framework, it's not a process or a methodology.
- Scrum is not just for software development but for the development of *any product*.

My goal is to get you Scrum Master certified. A *lean approach* to doing so is to not waste your time discussing peripheral topics that will distract and confuse you.

## Keep the Focus on the Goal

One of the Scrum values is focus.

I'mma try my very best to keep you focused, not waste your time on topics that aren't heavily tested on the exam, and at the same time, constantly re-inforce topics that I know will be tested heavily.

That's how we're going to get you past the Scrum Master certification finish line!



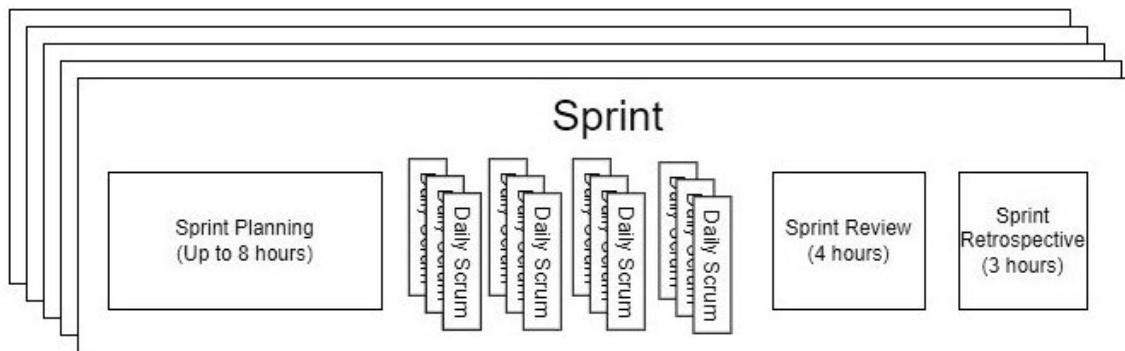
Figure 10. With lean thinking and focus, we're going to get you past the Scrum Master certification finish line. (Image: Joshbdork, GFDL1.2)



## 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 11. The Scrum pillars and values justify the need for the various Scrum events.*

## 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 pillars of Scrum:

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, with others 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) Transparency
- E) Adaptation

Options B, D and E 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 are 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 adapt accordingly.

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 one of the following problems will occur if product development lacks transparency?

- 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 why 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, which is why Scrum has become popular in those domains as well.

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.

The Scrum Guide describes several artifacts and timeboxed 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 your fellow 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 your fellow 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.

## **Commitment, Focus, Openness, Respect and Courage**

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
- D) 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.

## 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 12. A Scrum Team must have one Product Owner, one Scrum Master and some Developers to do the work. 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
- Engineers
- Architects
- Carpenters
- Heavy Equipment Operators 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
- Scrum Developers

**TIP**

You will often see questions on the Scrum Master certification exam where options include a reference to a *manager* or a *Project Manager*. The Scrum Guide never once mentions the terms *Project Manager* or *manager*. Options that refer to these positions are most likely wrong.

## There are no Subteams in Scrum

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

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

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.

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.

## Subteams are Radioactive 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 there is no performance team to do load testing. The development team does that.
- It means there is no team of technical writers to do documentation. 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 in the eyes of the Scrum framework.



Figure 13. 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)

## Test Yourself

Here's a prototypical Scrum exam question that tests you on 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.

## 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 or 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, your team better be actively figuring out 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.

How would you, the Scrum Master, respond? (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.

**TIP**

There is no reference to a 'Sprint 0', 'infrastructure Sprint' or 'release Sprint' in the Scrum Guide. Options that reference these concepts on the exam are typically wrong.

## 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 the impact.

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
- E) The stakeholders

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.

I will say that if I saw this question on an exam and 'Scrum Team' was an option, I'd lean towards answering 'Scrum Team' in favor of just the team of developers.

It's good to involve the entire team in important decisions.



Figure 14. A cross-functional Scrum Team has all the skills needed to complete a Product Backlog item from beginning to end. (Credit: Pierre Selim)

## Why only 10 to a Scrum team?

A Scrum Team's maximum size, 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.



*Figure 15. One person may wear different hats and assume multiple roles on a Scrum Team.*

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.

## 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) At least 5 people
- E) 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
- E) Small teams have lower maintenance costs

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
- The same Product Backlog
- The same 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.

## Keep Your Teams Small

Another takeaway is that Scrum advises you to break up your teams if they become too big, and that doesn't necessarily 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 incorrect 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

## End-to-End Feature Development

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.

## 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) An 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

The best way for the organization to empower a Scrum Team is to leave them alone and allow them to manage themselves. That's an important concept in Scrum. If the organization isn't committed to Scrum, the Scrum Team will inevitably encounter needless impediments that the Scrum Master will have to resolve.

## 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 useful Increment in 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.

## 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.

The Scrum Guide defines 3 accountabilities, not roles.

Scrum defines three specific accountabilities within the Scrum Team:

- The Developers
- The Product Owner
- The Scrum Master

— 2020 Scrum Guide page 5

## Linguistic Gymnastics

Throughout this book I hope you'll allow me to use the terms 'roles' and 'accountabilities' interchangeably. 'Roles' just sounds more natural, and I want this book to be approachable and easy to read.

I want everyone to love Scrum, and being incredibly strict with terminology and semantics can sometimes alienate people new to the framework. I think I'm pretty strict with terminology in this book, but I also give myself a bit of lee-way in the name of readability.

I will say, if you ever post on a Scrum forum or LinkedIn and refer to one of the Scrum accountabilities as a role, you'll get roasted.

There are a lot of gatekeepers in the community who treat the Scrum Guide as if it were scripture

written by men who were spoken to by God, and if you deviate at all from the word as it is written, you must be punished.

For example, in this book, I have called Scrum an incomplete framework.

However, technically, the Scrum Guide says "The Scrum Framework is purposefully incomplete."

Many Scrum zealots have challenged me on LinkedIn for saying that Scrum is an 'incomplete framework,' insisting that it is instead a 'framework that is *purposefully incomplete*,' and that the two statements are not the same.

Personally, I find these types of linguistic gymnastics to be very unhelpful, as they only push people away from Scrum. I want everyone to embrace it.

So just keep in mind when you are posting online, there will be people out there ready to pick you apart for using the colloquially accepted term 'role' instead of the technically correct term 'accountability.'

Choose your words wisely!

**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.



# The Developers

Scrum has its roots in software development, but the stewards of the framework have worked hard to push Scrum into other domains including:

- Construction
- Manufacturing
- Engineering
- Bee Keeping

To pass the Scrum certification exam, you must shake yourself out of the mindset that Scrum is all about software development.

A big mental transition many software developers have difficulty with is realizing that in terms of Scrum, a 'developer' isn't necessarily a 'software developer.'

Developers are the people in the Scrum Team that are committed to creating any aspect of a usable Increment each Sprint.

— 2020 Scrum Guide page 5

## Who is a Developer?

According to Scrum, the term 'developer' refers to anyone on the team who is adding value and working towards the creation of a usable and valuable increment of work.

- In construction, a developer could be the person painting the walls
- For castaways on a deserted island, a developer could be the person digging a trench for irrigation
- In beekeeping, a developer might be the person who disposes of the slumgum

As you read this section about the 'developer', it is easy to fall into the trap of thinking that Scrum is specifically talking about a 'software developer.' They're not.

Quite often, framing a scenario or exam question outside the world of software development will help you see things more clearly and choose the correct answer.

## Feature Development vs. Overall Accountability

The Scrum Certification exam will likely ask you one of these two questions:

- Who is accountable for creating a useful Increment?
- Who is committed to creating *aspects* of a usable Increment?

The two questions sound very similar, and if you're unprepared, the nuance between the job of the Scrum Developers and the duties of the Scrum Team might lose you an exam mark.

The Scrum Team as a whole is accountable for creating a useful Increment, not just the Scrum Developers.

However, it is the developers who create the various features, or **aspects**, of the usable Increment.

Compare the two quotes below and notice the difference the Scrum Guide makes between the Scrum Team and the Developers when it comes to the Increment:

The entire Scrum Team is accountable for creating a valuable, useful Increment every Sprint.

— 2020 Scrum Guide - Scrum Team Section

Developers are the people in the Scrum Team that are committed to creating any aspect of a usable Increment each Sprint.

— 2020 Scrum Guide - Developer Section

The whole point of Scrum is to deliver value to the stakeholders, and it is the developers who create all of the parts of the Increment, which is the additive unit of value in Scrum.

## The Developer's Job is Simple: Create

Despite the importance of the developer to the Scrum framework, the Guide only dedicates 84 words to them. It's because their job is quite simple - just create the product.

The remaining verbiage about developers outlines their key accountabilities:

The specific skills needed by the Developers are often broad and will vary with the domain of the work.

However, the Developers are always accountable for:

- Creating a plan for the Sprint, the Sprint Backlog;
- Instilling quality by adhering to a Definition of Done;
- Adapting their plan each day toward the Sprint Goal; and,
- Holding each other accountable as professionals.

— 2020 Scrum Guide - Developer Section

## Developer Accountabilities

Let's look at the accountabilities of the Developer one point at a time.

Developers are always accountable for creating a plan for the Sprint: the Sprint Backlog.

— 2020 Scrum Guide - Developer Section

There is only every one *Product Backlog*, although there are two backlogs discussed in the guide:

- One is the *Product Backlog* owned by a Product Owner. It outlines the work necessary to build the product.
- The other is the *Sprint Backlog* which is owned by the developers. The Sprint Backlog is a subset of the Prouduct Backlog.

The Sprint Backlog is the developer's plan of what they intend to get done during the Sprint. It is made up of items selected from the Product Backlog.

The developers own it, and nobody tells them how to manage it, not even the Scrum Master.

## Test Yourself

Who on the Scrum Team has the right to add new items to the Sprint Backlog? (Choose 1)

- A) The Scrum Master
- B) The Product Owner
- C) The Stakeholders
- D) The Developers

The answer is D.

Only the Developers have the authority to add or remove items from the **Sprint** Backlog.

In contrast, only the Product Owner can add or remove items from the **Product** Backlog.

## Instilling Quality

Developers are always accountable for Instilling quality by adhering to a Definition of Done.

— 2020 Scrum Guide page 5

Notice the connection between *quality* and the *Definition of Done*. Any time the Scrum Master certification asks how to enforce quality, the answer will get mapped to the Definition of Done.

## Test Yourself

The gatekeeper of quality for the Scrum Framework is: (Choose 1)

- A) The Scrum Master
- B) The Product Owner
- C) The Product Goal
- D) The Definition of Done

The answer is D. The Definition of Done is the quality gate all increments must pass through to become valid.

**NOTE** The Scrum Guide uses a maddening set of rules for capitalization. They always capitalize 'Developer.' I just can't do that. I capitalize 'Scrum Developer', but other references to developer I leave lower case.

## Daily Adaptation

Developers are always accountable for adapting their plan each day toward the Sprint Goal.

— 2020 Scrum Guide page 5

It is an assumption that things will change and plans will go awry during a Sprint. Developers are expected to constantly adapt their plans and adjust to changing conditions.

Since the developer's plan is the Sprint Backlog, that means the developers will be tinkering with, adding, deleting, and managing their Sprint Backlog constantly throughout the Sprint.

## Test Yourself

When are developers allowed to adapt their plan by changing the Sprint Backlog? (Choose 1)

- A) Any time during the sprint
- B) Only during Sprint Planning
- C) Only during the Daily Scrum

Option A is correct.

If the developers see an opportunity to adapt to changes, they are encouraged to update their plans immediately. In Scrum, the Sprint Backlog is the developer's plan. The developers can update the Sprint Backlog anytime during the Sprint if they feel it is necessary.

## Interpersonal Accountability

Developers are always accountable for holding each other accountable as professionals.

— 2020 Scrum Guide page 5

You are guaranteed to get a question on the Scrum Certification exam about who holds the developers accountable for monitoring or tracking their progress.

The answer is the developers. The developers hold each other accountable.

## The Role of the Scrum Master

People new to Scrum often think the 'Scrum Master', by virtue of the official sounding name, holds the developers accountable for their work.

That's not the case. The Scrum Master is really just a coach and facilitator.

It might also be understandable if you thought the whole team keeps the Scrum Developers accountable.

Again, that's not the case, which makes sense, because the Scrum Master or the Product Owner might not interact with the developers every day. On the other hand, the developers interact with each other at least once during the Daily Scrum.

In Scrum, the developers hold each other accountable for what they do and how they perform.

## What's not said about Developers

About 90% of the questions on the Scrum certification exam come directly out of the Scrum Guide. However, there are a few questions that require some additional knowledge.

Two topics you'll want to familiarize yourself with in regards to developers on a Scrum Team are:

- The fact that adding new developers can temporarily slow down a team's velocity
- The fact that new developers can be added any time throughout the Sprint

## Adding New Developers

When should you add new developers to a Team?

- During Sprint Planning?
- During the Sprint Review?
- During the Sprint Retrospective?

The Scrum Guide doesn't say anything decisive about this subject. It doesn't need to.

You can add new developers to a project any time you need them.

If you're halfway through a Sprint, and Human Resources just hired the four new programmers you've wanted for months, it wouldn't make sense to just keep those developers on the bench for two weeks so they can 'officially start' at the beginning of the next Sprint.

That would be silly and wasteful. Scrum is lean, empirical and when it makes sense, pragmatic.

## Sustainable Development

When should new developers be added to a team?

Scrum says developers should always work at a sustainable pace. Any time the pace starts to become unsustainable, add some new developers to the team.

Questions on the Scrum Master Certification exam might talk about budget money or internships or something like that, but those factors don't have anything to do with Scrum. Those topics are just put into questions to throw you off.

In terms of Scrum, developers can be added to a project any time they are needed. They can be removed at any time too.

## Test Yourself

Who holds the Scrum Developers accountable for their work?

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

The answer is D.

The Scrum Developers on the development team holds each other accountable for their actions.

## The Impact of Adding Developers

One reality of adding new developers to a team is that it temporarily slows the rest of the team down.

When new developers are onboarded, they usually take other developers away from their work, as the new developers are given help setting up printers, finding out where the washrooms are,

learning about the project and figuring out how to connect to GitHub.

That's just the reality of adding new people to a team.

For the short term, the productivity of individual members on the team will dip. Over time, individual productivity will go back to normal.

Just be aware of that fact on the Scrum certification exam.



*Figure 16. A new log thrown on a hot campfire will temporarily lower the fire's temperature. The same holds true for a new developer added to a hot project.*



# The Product Owner

It all starts with the Product Owner (PO).

Seriously, the whole idea of Scrum is for a team to work toward the creation of a product that makes stakeholders and customers happy.

- Who knows what the product's supposed to do?
- Who knows what features the product needs?
- Who has the product vision?
- Who knows what features matter the most?
- Who can quickly answer questions about the product?
- Who do the stakeholders trust to make decisions about the product?

The answer to all of those questions is 'The Product Owner.'

Who is completely useless in terms of knowing *how* to build a product, *estimating* completion times or even *decomposing* product features into easily manageable chunks? That's probably the Product Owner too.

The Product Owner (PO) knows the product. Just be careful never to ascribe more competency to the PO than they deserve. Assume they know nothing about actually building a product. That's not what they're accountable for according to Scrum.

## Product Owners Maximize Value

The Product Owner knows everything there is to know about the product the team's trying to build.

The Product Owner is accountable for maximizing the value of the product, resulting from the work of the Scrum Team.

How this is done may vary widely across organizations, Scrum Teams, and individuals.

— 2020 Scrum Guide page 5

The Product Owner knows all about the Product, but what is the Product Owner most accountable for? The Scrum Guide makes the Product Owner's job clear: to maximize the value of the product being created.

Interestingly, the Scrum Guide doesn't make it clear how the Product Owner goes about doing that. Fortunately, I can help you out here.

## How do Product Owners Maximize Value?

The PO maximizes the value of the work done by the Scrum Team by convincing the developers to work on the most important and valuable features of the product first.

- If the PO gets the developers to work on a bunch of features nobody is going to use, the PO is *not* maximizing value.
- If the PO gets the developers to create features the clients are dying to get their hands on, the PO is maximizing value.

That's how the PO maximizes the value of the stuff the developers are delivering.

## Product Backlog Management

Effective Product Backlog management is the key to maximizing the work done by developers.

The Product Owner is also accountable for effective Product Backlog management, which includes:

- Developing and explicitly communicating the Product Goal;
- Creating and communicating Product Backlog items;
- Ordering Product Backlog items; and,
- Ensuring that the Product Backlog is transparent, visible, and understood.

— 2020 Scrum Guide page 6

Take these four bullet points seriously, because Product Backlog Management is one of the 13 categories in which the questions on the Scrum certification exam are categorized. Furthermore, from what I've seen, this category is more heavily weighted on the exam than other categories.

There is a great deal of nuance to the responsibilities, or *accountabilities*, of the Product Owner. The exam will test your understanding of Product Backlog management by offering false statements or providing incorrect options that seem to make plausible sense.

## Product Owner Accountabilities

For example, the exam will assert in a True or False question that the *stakeholders* or *customers* are responsible for communicating the Product Goal to the team.

On the surface, that almost sounds like it makes sense. After all, the product is for the customers, right? So shouldn't they communicate the Product Goal?

Don't fall for the trick. The Product Owner communicates the Product Goal, not the stakeholders or the customers. The same trick goes for communicating Product Backlog items (PBIs).

It sounds logical that a customer or stakeholder might be the one creating or communicating details about Product Backlog items, after all, the Product Backlog represent features and

functionality that the customers want, right?

Again, don't fall for it. It's the Product Owner's job to maintain and communicate details about Product Backlog items to the team.

When it comes to the Product Backlog and the Product Goal, the responsibility falls on the shoulders of the Product Owner, nobody else.

## Test Yourself

Who is responsible for managing the Product Backlog and maximizing the value of the product produced? (Choose 1)

- A) The Scrum Master
- B) The Product Owner
- C) The management team
- D) The development team as a whole

Option B is correct.

As we just learned, the Product Owner manages the Product Backlog, and it's also the Product Owner's job to 'maximize the value of the product resulting from the work of the Scrum Team.'

## Test Yourself

Which of the following statements is true? (Choose 1)

- A) The Scrum Master can add items to both the Sprint Backlog and the Product Backlog
- B) The Product Owner can add items to both the Sprint Backlog and the Product Backlog
- C) The Product Owner is fully accountable for all items added to the Product Backlog
- D) The Scrum Master is fully accountable for all items added to the Sprint Backlog

Only Option C is correct.

The developers have full control over the **Sprint** Backlog and only the developers can add to it. The Product Owner has full control over the **Product** Backlog and only the PO can add items to it.

Notice in the question above the term 'accountable' was used. The Product Owner can delegate any of the work they are accountable for to anyone they deem fit. However, the Product Owner remains fully accountable for the results, regardless of who does the work.

## Test Yourself

It is the job of the Scrum Master to ensure that items in the Product Backlog are properly understood by both stakeholders and team members.

- True
- False

This is false. It is stated quite clearly in the Scrum Guide that it is the job of the *Product Owner* to make sure "that the Product Backlog is transparent, visible and *understood*."

Nobody should be expected to 'get into the head' of the Product Owner and assume what they are thinking.

While the Product Owner's vision may not always be complete, what vision they do have should be clear to everyone involved in the project.

## Who Does the Product Owner's Work?

In Scrum, the developers do the development. In Scrum, the Scrum Master does the Scrum Mastering.

Yet according to the Scrum Guide, the Product Owner can get someone else to do the gruntwork for them, just so long as they remain responsible and accountable for the final results.

The Product Owner may do the Product Backlog management work, or they may delegate the responsibility to others. Regardless, the Product Owner remains accountable.

— 2020 Scrum Guide page 6

## Product Ownership and Trust

Perhaps the most important quality of the Product Owner is that they have the complete and total trust of the organization to make decisions about what's best for the product being built.

For Product Owners to succeed, the entire organization must respect their decisions.

— 2020 Scrum Guide page 6

The importance of PO trust can't be understated with regards to product development.

- Scrum teams move quickly
- Conditions change from day to day
- Adaptation should happen constantly
- Product Backlog item clarity is of utmost importance

When developers have questions about the work they're doing, they need answers that are clear and definitive. The Product Owner must be able to respond quickly, and the development team needs to know that the decisions of the Product Owner will not be second-guessed by management.

If someone else in the organization keeps overriding the decisions of the PO, the team will quickly lose trust in the Product Owner, and the whole Scrum framework falls apart.

## A Single Product Owner

The job of the Product Owner cannot be shared amongst multiple individuals. There can't be two co-Product Owners on a team.

As the Scrum Guide stated earlier, the Product Owner can delegate some of the Product Backlog management work to a cohort, but in the end, all of the accountabilities associated with being a Product Owner fall on the shoulders of only one person: the PO.

The Product Owner is one person, not a committee.

The Product Owner may represent the needs of many stakeholders in the Product Backlog.

— 2020 Scrum Guide page 6

Notice how the Scrum Guide states that the Product Owner may represent the needs of many stakeholders. Implied in that statement is that different stakeholders might have different interests about which features or enhancements should be prioritized.

When it comes to competing interests, the Product Owner is responsible for managing expectations and building a consensus amongst the stakeholders.

## Keeping Tabs on the Product Owner

Stakeholders will always be interested in how the product is progressing. They will constantly want to know what's been created and what the team will be working on next.

The three Scrum artifacts, namely the Product Backlog, Sprint Backlog, and inspectable Increments are how stakeholders get the answers to their questions about how the project is progressing.

The entire organization must respect the PO's decisions. These decisions are visible in two ways:

1. through the content and ordering of the Product Backlog
2. through the inspectable Increment at the Sprint Review

Those wanting to change the Product Backlog can do so by trying to convince the Product Owner.

— 2020 Scrum Guide page 6

## Test Yourself

Important decisions about the future direction of the product being built must be made by:  
(Choose 1)

- A) The Scrum Master
- B) The Product Owner
- C) The Product Owner along with a stakeholder committee
- D) Stakeholders or executive management

The answer to this question is B, the Product Owner.

The Scrum Guide states quite emphatically that product-based decisions are not to be made by a committee.

## Test Yourself

There are many stakeholder groups with many competing interests, including the priority of features, the cost of the project, the release date, and community outreach.

How does Scrum ensure the interests of each of these groups are recognized and respected?  
(Choose 1)

- A) Have the Scrum Master represent the interests of each stakeholder group
- B) Have multiple Scrum Masters, with a different Scrum Master assigned to each stakeholder group
- C) Have one Product Owner represent the interests of each stakeholder group
- D) Have multiple Product Owners, with a different Product Owner assigned to each stakeholder group

The answer is C.

There is only one Product Owner on a Scrum Team, and that one Product Owner represents the interests of all the stakeholders.

### NOTE

Notice how the Scrum pillars of transparency, inspection and adaptation always arise when talking about Scrum events, accountabilities and artifacts.

## Trust, Transparency, and the Product Backlog

The PO must have the complete and total trust of the organization concerning product ownership, but trust is a two-way street. If the Product Owner is to be trusted, the Product Owner must also be transparent about what they are doing.

So how does the Product Owner make their decisions transparent? They do so by making the Product Backlog visible and available to all stakeholders.

- Want to know what the Product Owner is building? Look at the Product Backlog.
- Want to know what the Product Owner has prioritized? Look at the Product Backlog.
- Want to know what the Product Owner wants to build next? Look at the Product Backlog.
- Want to know how features are described? Look at the Product Backlog.
- Want to know the vision and goal for the product? Look at the Product Backlog's Product Goal.

The Product Backlog is the key to making the decisions of the Product Owner transparent.

## Inspection of the Increment

If a stakeholder wants to know what needs to be done, they look at the Product Backlog. If a stakeholder wants to know what's being done, they look at the Product Backlog items that are part of the current Sprint's backlog.

At the end of every Sprint, a Sprint Review takes place where stakeholders inspect the Product Backlog items that have met the Definition of Done to produce a product Increment.

The Product Owner's commitment to visibility and transparency with regard to the Product Backlog is what builds confidence and allows management to have trust in the Scrum framework.

## Negotiating Product Features

Not everyone will agree with what should be built next, what features should be prioritized, or how product development should be managed.

- Stakeholders might disagree.
- The Scrum Master might disagree.
- The Developers might disagree.

Disagreement is expected, especially in a fast-moving environment where things change quickly.

If anyone wants to change the Product Backlog, update the Product Backlog, delete something from the Product Backlog, or add something to the Product Backlog, they go through the Product Owner.

When it comes to the Product, and the Product Backlog that describes everything that needs to be built, the Product Owner has full control.

## Test Yourself

How does the Product Owner ensure their decisions are transparent and open? (Choose 3)

- A) By making their decisions visible in the Product Backlog
- B) By allowing stakeholders to see how the Product Backlog has been ordered and prioritized
- C) By sending regular status updates to stakeholders.
- D) By scheduling weekly meetings between the Scrum Developers and stakeholders

Options A and B are correct.

Everyone who has 'skin in the game' is allowed access to the Product Backlog. This allows stakeholders, developers and every team member to know exactly what needs to be built.

The ordering of the Product Backlog also allows stakeholders to know which items have been given top priority, and which items are less likely to be developed in the near future.

## Test Yourself

The CEO has told you, the Scrum Master, that if a key feature isn't added to the product within the next three weeks, the project will be cancelled.

What action should you, the Scrum Master, take? (Choose 1)

- A) Add the feature as a Sprint Backlog item so developers can start working on it immediately
- B) Add a new item to the Product Backlog to represent the feature
- C) Cancel the Sprint and have the developers shift their focus to this new feature
- D) Inform the Product Owner and facilitate a conversation between the Product Owner and the CEO

Option D is correct.

If a change needs to be made to the Product Backlog, it's the Product Owner who does it.

If the CEO needs a feature prioritized, the CEO has to go through the Product Owner.

Nobody has the right to add features to the Product Backlog other than the Product Owner, and nobody can inform the developers about which items are the highest priority other than the Product Owner.

## Chapter 7: The Scrum Master

People new to Scrum often assume that the Scrum Master has a great deal of power over other members of the Scrum Team.

They don't.

Scrum Masters don't have much power at all.

Most of what a Scrum Master does revolves around coaching and educating people about the Scrum framework.

- The Scrum Master doesn't make decisions about the Product
- The Scrum Master doesn't tell the developers how to manage their work
- The Scrum Master can't make changes to the Product Backlog
- The Scrum Master can't make changes to the work planned for the Sprint
- The Scrum Master can't hire or fire people

Even when the Scrum Master does do something of substance, such as helping to *cause the removal of impediments* or acting as a *leader who serves*, they typically do this by educating others about how Scrum is *supposed* to work and coaching people about how to properly embrace the ideas expressed in the Scrum Guide.



Figure 17. The Scrum Master coaches others on how to implement Scrum in accordance with the principles and ideas outlined in the Scrum Guide.

## Scrum Requires a Scrum Master

Every Scrum Team needs a Scrum Master. Scrum Teams have one Scrum Master, one Product Owner and one or more developers.

The Scrum Master is accountable for establishing Scrum as defined in the Scrum Guide.

They do this by helping everyone understand Scrum theory and practice, both within the Scrum Team and the organization.

The Scrum Master is accountable for the Scrum Team's effectiveness.

They do this by enabling the Scrum Team to improve its practices, within the Scrum framework.

— 2020 Scrum Guide page 6

A Scrum Master does not have to be 100% dedicated to a single Scrum Team.

New Scrum Teams will likely need a great deal of guidance on how to implement Scrum, but as teams mature, they typically require less coaching and less facilitation. At that point, a Scrum Master will likely split their time across several teams.

A great Scrum Master aims to coach themselves into irrelevance.

## Test Yourself

The developers say they don't feel ready to start development and would like to do an infrastructure Sprint to get everything set up first.

However, the Product Owner wants to get the first Sprint, and the first increment, under the team's belt. What should the Scrum Master do? (Choose 1)

- A) Cancel the first Sprint.
- B) Tell the developers there will not be an 'infrastructure Sprint.'
- C) Tell the Product Owner there will be no Increment on the first Sprint.
- D) Help the Scrum Developers understand the importance Scrum places on creating an increment in every Sprint.

Option D is correct.

The job of the Scrum Master is to:

- Coach
- Facilitate
- Help everyone understand how Scrum works.

## What the Scrum Master Doesn't Do

People new to Scrum often ascribe too many responsibilities to the Scrum Master. They think the Scrum Master has a great deal of control over all aspects of a development project.

That's not true. A Scrum Master is the master of Scrum and its implementation. They are not a business consultant, they don't determine project timelines and they definitely don't manage a project's budget.

A Scrum Master does not:

- Manage the project
- Assign tasks to members of the team
- *Tell* developers what to do
- Cancel, delay, or extend Sprints
- Assign developers to teams

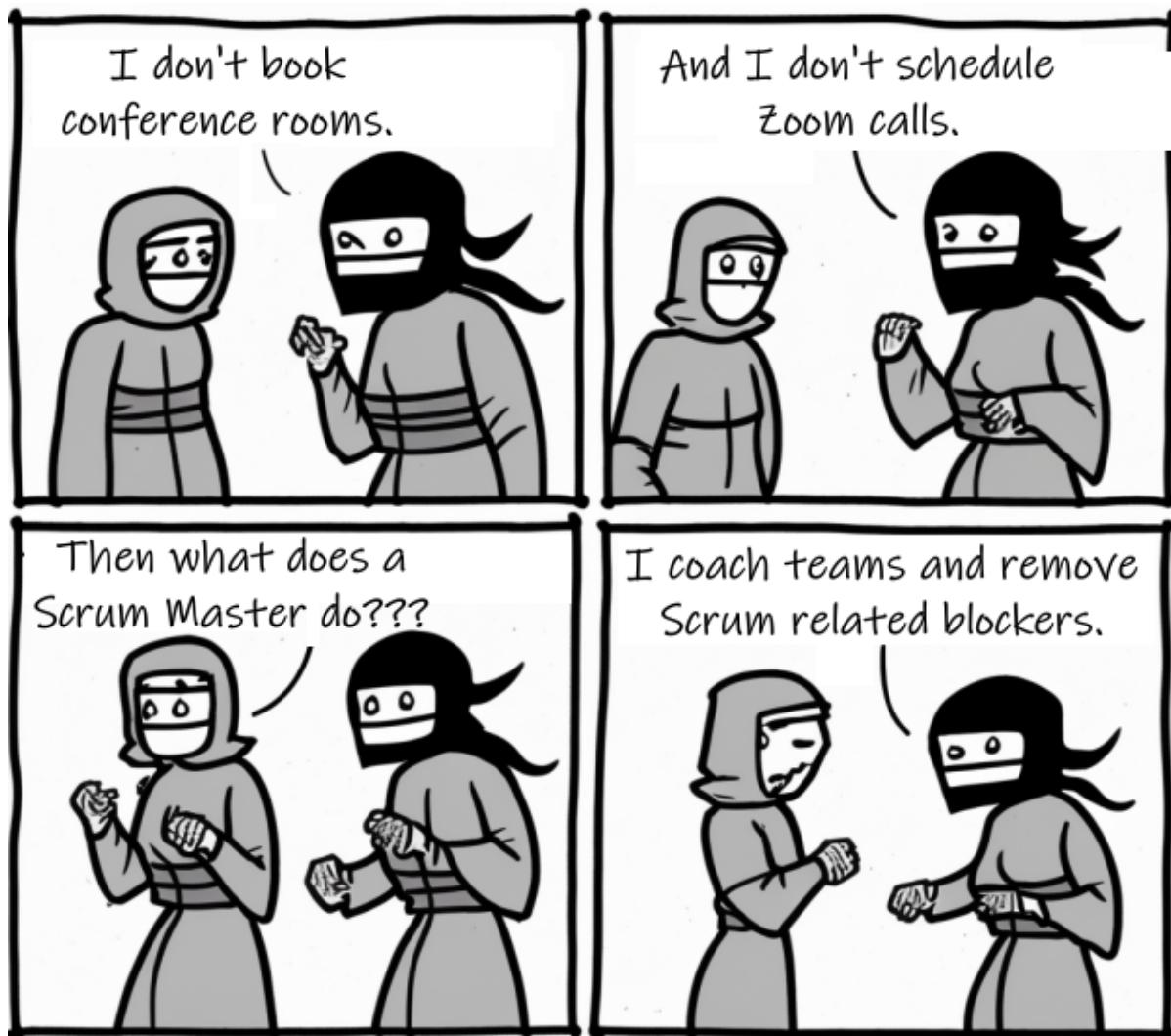


Figure 18. The Scrum Master facilitates, coaches and removes Scrum related impediments.

## Scrum Masters and Servant Leadership

Previous editions of the Scrum Guide talked about servant leadership. Yet the 2020 Scrum Guide removes all references to the term 'servant leader.'

Instead, the 2020 Scrum Guide includes this line that spins the idea of servant leadership around a bit:

Scrum Masters are true leaders who serve the Scrum Team and the larger organization.

— 2020 Scrum Guide page 6

The Scrum Guide always tries to be very precise in its words.

The stewards of the Scrum framework want everyone to know that a Scrum Master is a leader first, and then a servant to the team. They are a leader who serves, not a servant who leads.

It's a persnickety distinction, but an important one nevertheless.

## Test Yourself

Which one of the following statements is most true about a Scrum Master?

- A) The Scrum Master manages others with a command and control leadership approach
- B) The Scrum Master is a leader first, and a servant to their Scrum Team second
- C) The Scrum Master is a servant-leader on the Scrum Team
- D) The Scrum Master is a leader and not a servant

Option B is correct.

The Scrum Master is a leader first who also serves the team.

## How Scrum Masters Serve the Scrum Team

The Scrum Master serves the Scrum Team in several ways, including:

- Coaching the team members in self-management and cross-functionality;
- Helping the Scrum Team focus on creating high-value Increments that meet the Definition of Done;
- Causing the removal of impediments to the Scrum Team's progress;
- Ensuring that all Scrum events take place and are positive, productive, and kept within the timebox.

— 2020 Scrum Guide page 6

The first two bullet points emphasize the point made earlier in the Scrum Guide. Namely that the Scrum Master is there to help the team by coaching them on how Scrum work.

Notice the use of terms such as 'coaching' and 'helping' in the definition of the Scrum Master's *accountabilities*.

On the Scrum Certification exam, if you ever see an option that implies a Scrum Master will force somebody to do something, or *tell* somebody to do something, that option will typically be wrong.

A Scrum Master is:

- an educator
- a facilitator
- a coach
- a diplomat
- a leader who serves

They are not managers or commanders who coerce others through intimidation or Machiavellian tactics.

## Test Yourself

A developer is having a problem with some of the front end code. What should the Scrum Master do? (Choose 2)

- A) Tell the Developer what needs to be fixed in the code
- B) Coach the Developer on how to leverage the team for help
- C) Instruct the Developer to change the framework
- D) Facilitate a team based approach to providing the dev with support
- E) Force the Developer to take a different approach

Options B and D are correct.

There will be questions on the Scrum Certification exam where some of the options include the Scrum Master telling, instructing, insisting, or demanding a Developer does something. Answers like that will always be wrong and can be disqualified immediately.

A Scrum Master is a coach, facilitator, and a leader who serves. They are not army generals who command.

Answers that include terms like *coach* or *facilitate* are more in line with the accountabilities a Scrum Master is responsible for.

## Removal of Impediments

The Scrum Guide says the Scrum Master should work to cause the removal of impediments to the progress of the team.

The Scrum Master serves the Scrum Team by causing removal of impediments to the Scrum Team's progress

— 2020 Scrum Guide page 6

So what's an impediment?

It's important to keep in mind that we are talking about impediments within the context of the Scrum framework.

For example, if a manager starts asking for daily status reports from the developers detailing what they're working on, that infringes on the development team's ability to self-manage.

The demand to needlessly fill out status reports that zap the dev team's productivity is an impediment that the Scrum Master should work to remove.

In this case, the Scrum Master would coach management on the need to allow developers to manage themselves, while at the same time showing the managers how the Sprint Backlog and the Sprint Review are much better ways to gain insight into the progress of the development.

Similarly, if a manager is trying to attend the Daily Scrum, or the Product Owner is not properly communicating the details of their Product Backlog items, these are impediments the Scrum Master can work to remove.

## In the Context of Scrum

Impediments that fall outside the context of Scrum are not typically things the Scrum Master would be expected to solve.

- If a developer can't figure out how to book a conference room for the Daily Scrum, that's not exactly a blocker for the Scrum Master to solve.
- If the computer hosting the Git repository goes up in flames and the team needs a new computer, that's not exactly a blocker for the Scrum Master to solve.

The Scrum Master works to cause the removal of blockers that slow the team down, but only within the context of the Scrum Guide and the Scrum framework.

Removing blockers typically involves coaching, helping, and educating others about Scrum. It doesn't involve doing people's jobs for them. That's enablement, not impediment removal.

**TIP**

Avoid options on the Scrum Master exam that use terms like *tell*, *assign* or *reprimand*. Scrum Masters coach and facilitate. They don't *tell*, *assign* or *reprimand*.

## Test Yourself

A developer is trying to book a conference room for the Daily Scrum, but they can't figure out how to use the company's conference room booking system.

Which one of the following courses of actions should the Scrum Master take?

- A) The Scrum Master should remove the blocker by learning the booking system and booking the room for the developer
- B) Allow the developer to book a different room each day for the daily Scrum
- C) Coach the developer on the importance of Scrum Events always taking place at the same time and location
- D) Have the developer only schedule the Daily Scrum on days when the same room is available

Option C is correct.

The booking system may be a blocker or impediment to the developer, but it does not impede progress in terms of the Scrum framework. It's not the Scrum Master's job to teach a developer how to use their own tools.

Perhaps a Scrum Master could advise the developer on how to find training that would teach them how to better use the tool, or facilitate a learning session with other developers where knowledge sharing about the booking system could happen. But doing the developer's work for them is not what the Scrum Guide means when it talks about removing impediments.

In this case, the only correct option in the question is for the Scrum Master to coach the developer on the importance of having Scrum Events always take place at the same time and location, so long as it's reasonable to do so.

## Examples of Impediments a Scrum Master Can Resolve

The types of impediments a Scrum Master is expected to help solve are typically things that go beyond issues that self-managed and self-organized teams can solve on their own, while at the same time falling within the context of Scrum. Examples include:

- Availability of the Product Owner
- The team being pushed beyond a sustainable pace
- External attempts to micro-manage the team
- Unclear Sprint and Product Goals

All of these issues are impediments the Scrum Master should be able to help remove through coaching, facilitation, and educating people about the proper application of the Scrum framework.

## Positive, Productive, and Timeboxed

The final bullet point from the Scrum Guide emphasizes the Scrum Master's role in making sure all Scrum events happen throughout the Sprint, that they are productive, and that they are timeboxed.

The Scrum Master serves the Scrum Team by ensuring that all Scrum events take place and are positive, productive, and kept within the timebox.

— 2020 Scrum Guide page 6

When answering questions on the certification exam, remember that according to this point:

- It's the Scrum Master's job to ensure that all Scrum Events take place
- It's the Scrum Master's job to ensure that all Scrum Events fall within their timebox
- It's the Scrum Master's job to ensure all Scrum Events are positive
- It's the Scrum Master's job to ensure all Scrum Events take place at the same time
- It's the Scrum Master's job to ensure all Scrum Events take place in the same location

There will be questions on the exam that state the developers don't want to do the Daily Scrum every day, or they want to hold the Daily Scrum in different locations on certain days of the month.

In these situations, the Scrum Master must coach the team on the importance of taking part in all Scrum events and making sure they take place at the same time and in the same location every time.

There will also be questions about whose job it is to keep Scrum Events within their allotted time. In situations like these, it is the Scrum Master's job to coach the team about the importance of keeping within the timebox.

Other questions will say that developers are not enjoying the Daily Scrum, or they don't want to take part in the Sprint Review or Retrospective. In these situations, it's the Scrum Master's job to coach the team about the importance of these events and try to facilitate a strategy that will make the events more positive and rewarding for those who attend them.

## Test Yourself

The Scrum Developers are frustrated, not clear on what they are developing, and do not want to attend the Daily Scrum. What should the Scrum Master do? (Choose 2)

- A) Cancel the Daily Scrum so the developers can focus on their work.
- B) Coach the team on the value the Daily Scrum brings to the team.
- C) Reprimand developers who don't attend the Daily Scrum
- D) Facilitate the creation of a clearer Sprint goal during the next Sprint Planning event

B and D are correct.

The Scrum Master can't cancel any Scrum Events. The Scrum Master must coach the team to ensure *all Scrum Events take place*, are positive, and are productive, so option A is incorrect. The Scrum Master also doesn't have the authority to reprimand anyone, so option C is incorrect.

The Scrum Master should coach the team on the value of the Daily Scrum, and also help the team create clearer goals so they have more focus on future Sprints.

---

## Scrum Master and the Product Owner

The Scrum Master serves the Product Owner in several ways, including:

- Helping find techniques for effective Product Goal definition and Product Backlog management;
- Helping the Scrum Team understand the need for clear and concise Product Backlog items;
- Helping establish empirical product planning for a complex environment;

— 2020 Scrum Guide page 6

Notice how the term 'helping' is used repeatedly when describing how the Scrum Master serves the Product Owner. Furthermore, each item the Scrum Master is helping with is an element of Scrum, such as:

- The Product Goal
- The Product Backlog
- Empiricism

Helping the Product Owner with these things works to remove impediments for the team.

For example, poorly defined Product Backlog items make it difficult for the developers to understand what is required of them. Encouraging the Product Owner to create clear Product Backlog items removes an impediment.

## Product Owner Impediments

A poorly defined Product Goal means it's not clear what the Scrum Team's efforts are aiming at. By helping the Product Owner effectively define the Product Goal, another impediment to progress is removed.

A lack of empirical product planning can lead to budgetary problems, timelines getting overshot and development grinding to a halt because of unsatisfied dependencies on other teams.

These are all impediments to progress that can be resolved if the Scrum Master helps the Product Owner to more effectively do what the Scrum Guide requires of them.

---

## The Scrum Master and the Stakeholders

Many people approach Scrum with the assumption that the framework employs some type of command and control structure that rigidly defines hierarchies and sets boundaries around who can talk to each other and when.

Nothing could be further from the truth.

Developers, Scrum Masters, Product Owners, stakeholders, and even customers are encouraged to talk to each other. There are no rules in Scrum about stakeholders only talking to Product Owners, or developers not being allowed to talk to customers.

Scrum promotes the exact opposite type of approach. Scrum values openness and transparency.

One of the Scrum Master's main responsibilities is to remove barriers between the stakeholders and the Scrum Team, helping to open up channels of communication and enhance transparency in the process.

## Removing Barriers Between Stakeholders and Teams

Imagine a small startup where the office has 8 desks for the following people:

- The two owners of the startup, who are also the stakeholders in the product being built
- One Product Owner
- One Scrum Master
- Four developers

Do you think the stakeholders and the developers should never talk to each other in that situation?

Do you think there should be some rule that says the owners should put on noise-cancelling headsets any time two developers talk about the project?

Or do you think it would improve transparency if the developers and the stakeholders talked to each other all the time?

One of the jobs of the Scrum Master is to remove barriers between stakeholders and Scrum Teams. It's always preferable to have open and healthy lines of communication between everyone on a project.

The Scrum Master serves the organization by removing barriers between stakeholders and Scrum Teams.

— 2020 Scrum Guide page 6

Some people also get the incorrect impression that developers can only talk to stakeholders during the Sprint Review.

The Sprint Review is a scheduled event that makes sure the stakeholders and the Scrum Team get a chance to talk, but it is not the only time such communication is allowed.

Openness and transparency are always better than secrecy and ambiguity.

## Planning and Advising

The Scrum Guide doesn't speak much about activities that happen outside of a Sprint. However, the Scrum Guide does say the Scrum Master is expected to help an organization get up and running with Scrum by helping to plan the adoption of Scrum.

The Scrum Master serves the organization by planning and advising Scrum implementations within the organization

— 2020 Scrum Guide page 6

Along with helping to get Scrum up and running, the Scrum Master is also expected to help the organization as a whole by coaching others about how Scrum works and how Scrum's empirical approach to problem-solving can aid in product development.

The Scrum Master serves the organization in several ways, including:

- Leading, training, and coaching the organization in its Scrum adoption;
- Helping employees and stakeholders understand, and enact an empirical approach for complex work;

— 2020 Scrum Guide page 6

The job of the Scrum Master is to teach people how to do Scrum, how to implement Scrum, and how to improve on Scrum-based practices.

## Test Yourself

The management wants regular status updates on the progress of the project, so they have scheduled a standup meeting every Friday afternoon.

How should you, as the Scrum Master, handle this situation?

- A) Tell the development team they need to attend a Friday afternoon standup meeting
- B) Replace Friday's Daily Scrum with management's standup meeting
- C) Have the Product Owner speak to the management about the need for the Friday meeting
- D) Speak personally to the management about the need for the Friday meeting

Option D is correct.

In this case, there is an issue between the Stakeholders and the Scrum Team.

Hopefully, management can be coached on the transparency built into Scrum, and gain enough value from the Sprint Review to remove the request for additional meetings with the Scrum Teams.

Also, by keeping the Scrum Developers out of an unneeded meeting, the Scrum Master has removed an impediment to the Scrum team.



*Figure 19. The Scrum Master is a leader who serves a team of peers and equals.*

## Chapter 8: Scrum Events

The are four events in Scrum that occur within the confines of a Sprint:

- The Sprint Planning event
- The Daily Scrum
- The Sprint Retrospective
- The Sprint Review

The Sprint is a container for all other events.

— 2020 Scrum Guide page 7

### The Sprint Container

The Scrum Guide says the Sprint is a container for all other events. That means the Sprint Review, Sprint Planning, the Sprint Retrospective, and the Daily Scrum all happen within the Sprint.

- None of the Scrum events happen before the Sprint.
- None of the Scrum events happen after the Sprint
- Nothing in Scrum happens at the moment where one Sprint ends and the next Sprint begins

A new Sprint starts immediately after the conclusion of the previous Sprint.

All the work necessary to achieve the Product Goal, including:

- Sprint Planning,
- Daily Scrums,
- The Sprint Review, and;
- The Sprint Retrospective, happen *within* Sprints.

— 2020 Scrum Guide page 7

## Test Yourself

The team wants to update their Definition of Done. When is the right time to make this change? (Choose 1)

- A) Any time after the current Sprint ends but before the next Sprint starts.
- B) During the current Sprint's Sprint Retrospective.
- C) Immediately after the current Sprint ends, but before the next Sprint begins.
- D) Any time between Sprints.

The correct answer is B.

Since nothing happens before or after a Sprint, all options other than B are incorrect.

The Sprint Retrospective is the right time to update the team's Definition of Done.

Don't be caught off guard by the fact that this question is talking about the Definition of Done (DoD) or the Sprint Retrospective, topics we don't cover until a little later. The question isn't testing you on your knowledge of the Definition of Done. It's testing you on your understanding of how the Sprint works.

## The Importance of the Scrum Events

The pillars of Scrum are transparency, inspection and adaptation. The Scrum events are like feedback loops that happen at regular intervals which allow inspection and adaptation to happen.

For the inspection that happens and the adaptations that result to be beneficial, full transparency with regards to the team's progress, which is shown through Scrum artifacts like the Increment and the Product Backlog, is required.

Each event in Scrum is a *formal* opportunity to inspect and adapt Scrum artifacts.

These events are specifically designed to enable the transparency required. Failure to operate any events as prescribed results in lost opportunities to inspect and adapt.

— 2020 Scrum Guide page 7

Inspection and adaptation are important concepts in the Scrum framework. The Scrum events provide scheduled opportunities for that to happen, but you should be prepared to adapt to change at any point in time.

Any time a Scrum Team recognizes a change that demands a response, the team should respond and adapt immediately. If it's possible to adapt and change the plan prior to a scheduled event, the team should do so.

## Test Yourself

Only the Sprint Review provides the opportunity for the Scrum Team to inspect and adapt.

- True
- False

This is false.

All of the Scrum events provide the opportunity for the team to inspect their work and adapt their plans.

Furthermore, the Scrum Team is encouraged to inspect and adapt constantly.

There is never a bad time to adapt your plan if it helps your team achieve the Sprint Goal or come closer to achieving the Product Goal.

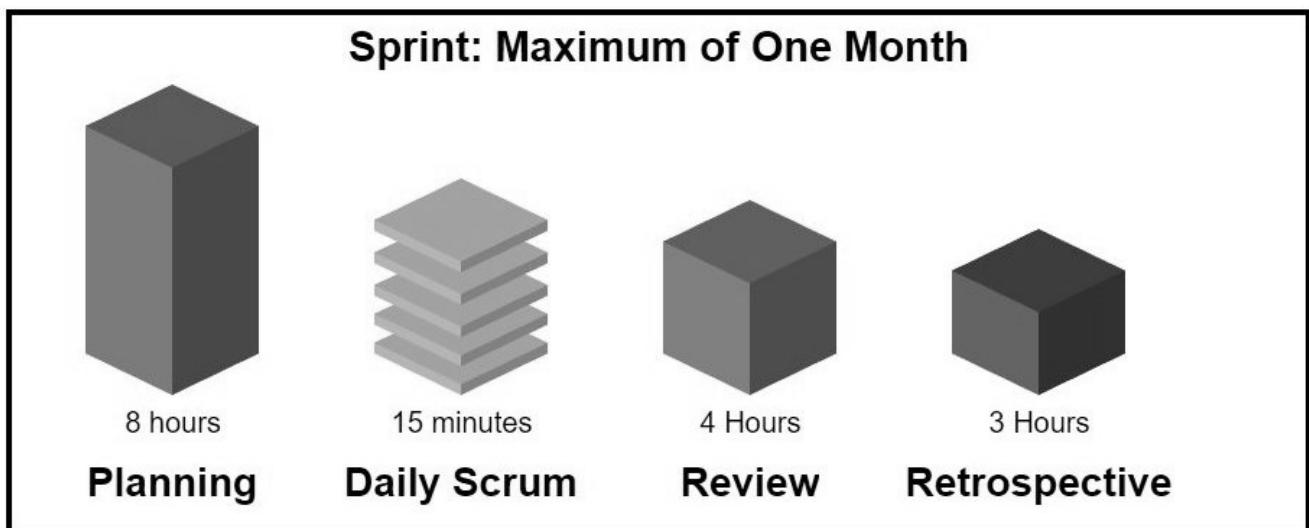


Figure 20. The four Scrum events all take place within a fifth event known as the Sprint. Each event is timeboxed, which means it cannot exceed the maximum amount of time allotted to it.

## Regularity and consistency

Scrum doesn't place restrictive controls over how development takes place, but it does request that certain patterns be followed, and it requires that those patterns happen with predictable regularity.

Events are used in Scrum to create regularity and to minimize the need for meetings not defined in Scrum.

— 2020 Scrum Guide page 7

## Every Scrum Event Serves a Purpose

Many people complain that Scrum has too many meetings. It doesn't.

- There is one timeboxed meeting to plan what to do during the Sprint
- There's a quick, 15-minute Daily Scrum to allow developers to touch base
- There is a Sprint Review where the team shares their progress
- There's a Sprint Retrospective where the team talks about what they could do better

That doesn't sound like too many meetings to me.

The meetings that Scrum has are intended to have maximum effectiveness on transparency, while at the same time eliminating the need for other meetings.

Teams that complain about Scrum having too many meetings probably aren't doing Scrum properly. Scrum has exactly the right number of meetings.

### Test Yourself

The CTO wants to schedule afternoon status meetings with the dev team to monitor the progress of an important feature. What should you, the Scrum Master, do? (Choose 1)

- A) Allow the afternoon meetings to take place until the feature is complete
- B) Invite the CTO to participate in the Daily Scrum in place of the afternoon meetings
- C) Coach the CTO on how Scrum provides transparency through existing Scrum events and artifacts
- D) Have the Product Owner send reports from the Daily Scrum to the CTO to avoid the extra meetings

Option C is correct.

When stakeholders attempt to manage the Scrum Developers, it is often because they want more transparency into what is happening in terms of product development.

Scrum already provides enough meetings, in the form of events, to allow for transparent inspection of progress. Scrum artifacts like the Product Backlog and the Increment provide further transparency. If a stakeholder is concerned about transparency, a Scrum Master should coach them on how to use existing events and artifacts to get the transparency they need.

#### WARNING

Every event in Scrum is required. You can never allow any Scrum event to be skipped. If you skip any events, you're not doing Scrum.

## Same Time, Same Place

While it's not always pragmatically possible, Scrum asks that all of the Scrum events happen at the same time and in the same place.

- Sprint Planning should always take place at the same time and location
- The Daily Scrum should always take place at the same time and location
- The Sprint Review should always take place at the same time and location
- The Sprint Retrospective should always take place at the same time and location

This makes the Scrum events more predictable, easier to plan around, and more likely to have full participation from everyone involved.

Optimally, all events are held at the same time and place to reduce complexity.

— 2020 Scrum Guide page 7

Of course, Scrum isn't completely unreasonable and unforgiving when it comes to a rule like this. The time and place are allowed to be adjusted for pragmatic reasons.

I mean, if the place you typically hold the Daily Scrum is being fumigated, it's okay to move the Daily Scrum to Conference Room B. Just try to keep the time and place as unchanged as possible.

## Test Yourself

The development team wants to move Friday's Daily Scrum, which normally takes place at 1pm, to 8 am so developers can leave work early for the weekend.

How do you respond as the Scrum Master? (Choose 1)

- A) Respect the self-managing Scrum Team and reschedule Friday's Daily Scrum
- B) Change the time of the Daily Scrum to 8am for every day of the week
- C) Explain to the team that the Daily Scrum must always take place at the same time and location
- D) Ask the Product Owner if it's agreeable to changing the Daily Scrum to 8 am on Friday

Option C is correct.

Since the Daily Scrum is a Scrum Event, and since Scrum Events are supposed to take place at the same time and location every time they occur, the Scrum Master would need to explain to the team that it can't arbitrarily change the time the Daily Scrum takes place on Fridays.



## Chapter 9: The Sprint

All of the development, planning, retrospectives, and reviews that happen in Scrum happen within the scope of a Sprint. It's the most important part of Scrum.

Sprints are the heartbeat of Scrum, where ideas are turned into value.

— 2020 Scrum Guide page 7

### Fixed Length Sprints

A Sprint has a fixed length of one month or less.

Not to sound pedantic, but 'fixed' means the length of a Sprint can't be changed once it's been started.

Sprints are fixed-length events of one month or less to create consistency.

— 2020 Scrum Guide page 7

Sometimes the Scrum Certification exam will ask you when it's possible to change the length of a Sprint.

The temptation is to say 'during Sprint planning,' but Sprint planning happens *during* the Sprint, so if you were to change the duration of the Sprint during Sprint Planning, it would be the length of the *next* Sprint you'd be changing, not the current one.

Discussions about the length of the Sprint and whether the Sprint should be shortened or extended are best to take place during the Sprint Retrospective. That way, if you do decide to change the Sprint length, the change can apply to the Sprint that starts after the current Sprint Retrospective finishes.

Having said that, the length of the Sprint shouldn't be changed too often. It should remain relatively consistent throughout the project.

### One Month or Less

Another key point here is the fact that a Sprint is a month or less.

- It's not 28 days
- It's not 30 days
- It's not 4 weeks

The length of a Sprint is one month or less. You'll be tested on that, so commit it to memory.

## The Start of the Sprint

A new Sprint starts immediately after the conclusion of the previous Sprint.

— 2020 Scrum Guide page 7

We've already discussed how the next Sprint starts immediately after the current Sprint ends.

The Scrum Guide re-states this point again here, which is a good clue for you to pay attention to it. You'll be tested on this fact several times on the exam.

On page 10 of the Scrum Guide, more clarification is given about exactly when the Sprint ends:

The Sprint Retrospective concludes the Sprint.

— 2020 Scrum Guide page 10

This means there are technically two correct answers when asked when a new Sprint begins:

1. A new Sprint begins as soon as the previous Sprint finishes.
2. A new Sprint begins when the Sprint Retrospective ends, since the Sprint Retrospective concludes the Sprint.

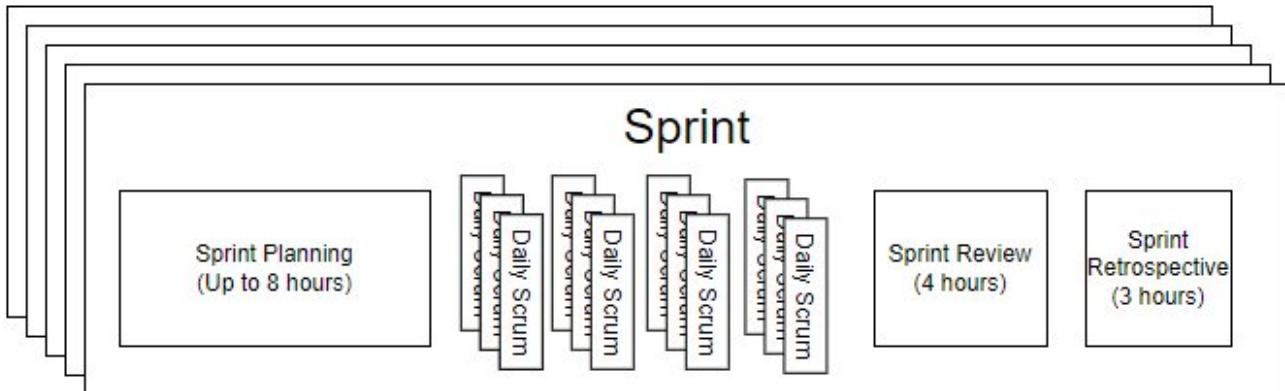


Figure 21. All of the Scrum events happen within the scope of a Sprint. When one Sprint ends, a new immediately begins.

## Test Yourself

Sprint Zero can last for up to 49 days.

- True
- False

This is false for two reasons.

First, there is no such thing as 'Sprint Zero' as far as the Scrum Guide goes.

- There are no Release Sprints in Scrum
- There are no UAT Sprints in Scrum
- There are no QA Sprints in Scrum
- There are no Infrastructure Sprints in Scrum
- There are no Sprint Zeros in Scrum

Whenever you see the term 'Sprint Zero' on the Scrum Master Certification exam, it's usually a red herring.

Secondly, this statement is false because the maximum length of a Sprint in Scrum is one month. That's the max. Often it is shorter.

---

## What Happens During the Sprint?

It's always assumed that things will change unpredictably from day to day during a Sprint. This is why Scrum advocates constantly inspecting and adapting.

All the work necessary to achieve the Product Goal, including Sprint Planning, Daily Scrums, Sprint Review, and Sprint Retrospective, happen within Sprints.

During the Sprint:

- No changes are made that would endanger the Sprint Goal;
- Quality does not decrease;
- The Product Backlog is refined as needed; and,
- Scope may be clarified and renegotiated with the Product Owner as more is learned.

— 2020 Scrum Guide page 7

Here are two things that are not allowed to change during the Sprint:

1. No changes are made that put the Sprint Goal at risk
2. No changes are made to decrease the quality of the existing product

The Scrum Guide doesn't have many hard and fast rules, but those are two of them.

**TIP** Some organizations declare a *Sprint Zero* at the start of the project to get the team organized. This is not a Scrum endorsed concept. When mentioned on the exam, it is typically an incorrect option.

## Test Yourself

During development, the Scrum Team has found a selected Product Backlog item is much more complex than initially anticipated.

The developers believe this PBI should be broken into multiple Product Backlog items that should be scheduled across multiple Sprints. What should you as the Scrum Master do? (Choose 1)

- A) Have the Scrum Developers clarify the requirements and renegotiate the scope of the Product Backlog items with the PO
- B) Expand the length of the Sprint so the Development team can complete the complex Product Backlog items
- C) Rewrite the user story so the work related to the Product Backlog Item can be completed within the current Sprint

A is correct.

If the developers have issues with the size, scope, complexity, and clarity of their work, they simply contact the Product Owner so requirements can be clarified and scope can be renegotiated.

Sprints are a fixed length, so they can't ever be extended, which makes B incorrect.

And the Scrum Guide makes zero references to user stories, so any talk of user stories on the Scrum Master certification exam will likely be a red herring.

## Test Yourself

A Sprint can be any length a team decides is best, so long as it is less than a month.

- True
- False

This is true.

The Scrum Guide doesn't specify the minimum length of a Sprint. When the Scrum Guide doesn't explicitly define something, it's up to the Scrum Teams to figure things out for themselves.

I couldn't imagine fitting a Sprint Planning meeting, Sprint Retrospective, and Sprint Review into a 24-hour Sprint, but nothing in the Scrum Guide forbids it.

## Why do we have Short Sprints in Scrum?

Some people often wonder why Scrum emphasizes short sprints.

The answer is simple. The longer the Sprint, the less predictable things become.

It's hard enough to predict four days into the future, let alone four weeks.

Short Sprints make things more predictable. Short Sprints also allow more frequent interactions with stakeholders as Sprint Reviews, which occur at the end of the Sprint, take place more often.

Sprints enable predictability by ensuring inspection and adaptation of progress toward a Product Goal at least every calendar month.

When a Sprint's horizon is too long the Sprint Goal may become invalid, complexity may rise, and risk may increase.

Shorter Sprints can be employed to generate more learning cycles and limit the risk of cost and effort to a smaller time frame.

Each Sprint may be considered a short project.

— 2020 Scrum Guide page 7

## The Benefits of Short Sprints

The Scrum Guide places a great deal of emphasis on short Sprints, and there are plenty of reasons for that.

Remember, a Sprint Review happens at the end of a Sprint. With a 2 week Sprint, the stakeholders get to review the product twice in a month. With a 4 week Sprint, they only get to see it once.

A short Sprint makes it easier to manage risk. If the team is doing something wrong, they'll find out sooner from the stakeholders in a 2-week Sprint than one that lasts a month.

It's also arguably easier to estimate work for two weeks, as opposed to three or four, and the project is less likely to go completely sideways in two weeks as opposed to two months.

## Empiricism Trumps Scrum Metrics

Various practices exist to forecast progress, like burn-downs, burn-ups, or cumulative flows.

While proving useful, these do not replace the importance of empiricism.

In complex environments, what will happen is unknown. Only what has already happened may be used for forward-looking decision-making.

— 2020 Scrum Guide page 7

The Scrum Guide is light on metrics.

The only Scrum metrics referenced in the guide are mentioned here:

- Burn-down charts
- Burn-up charts
- Cumulative flows

What's more, the guide kinda throws shade on them, saying that while they are useful, they're not as good as empirical analysis, which is a bit of a diss.

Still, for the exam, you do need to know what each of these charts does. You don't need to understand them in depth, but questions will arise about what they can tell you.

## Test Yourself

The development team is working with a new technology which has introduced a great deal of risk and uncertainty into the project. The Product Owner wants to reduce the Sprint time down to 2 or 3 weeks from the current Sprint length of one month.

What should the Scrum Master do? (Choose 1)

- A) Act on the Product Owner's recommendation and reduce the Sprint length to 2 weeks
- B) Coach the Scrum Team on how shorter Sprints reduce risk and ask them to consider reducing the Sprint length to 2 or 3 weeks.
- C) Explain to the Product Owner that the Sprint length cannot be changed after development has started.
- D) Plan a week-long 'learning Sprint' where developers learn the new technology and don't try to deliver an Increment.
- E) Assign work with the new technology to an external Scrum Team that specializes in it.

The correct answer is B.

As a Scrum Master, your job is to teach people about Scrum practices and Scrum Theory and coach them on how to apply Scrum in a manner that works best for them. Neither the Scrum Master nor the Product Owner can arbitrarily set the length of a Sprint in Scrum. This is a decision that must come about through the consensus of the team.

C is incorrect because the length of future Sprint can be changed if the team agrees a change is needed. However, the Sprint length cannot be changed mid-sprint.

D is incorrect because there is no 'learning Sprints' in Scrum. Furthermore, it is a requirement that every Sprint works toward the creation of a valuable and useful Increment.

E is incorrect because Scrum Teams should be able to learn new technologies as required.

## Scrum Can't Predict the Future

Managers and team leaders always want metrics that will help predict when a given project will be finished, or when a product will hit a given milestone. Scrum is quick to downplay the various charting techniques Agile teams frequently use to predict the future.

The Scrum Guide emphasizes that there are limits to the accuracy of charts and metrics, and there is no replacement for empiricism.

While proving useful, these [charts and metrics] not replace the importance of empiricism.

In complex environments, what will happen is unknown.

Only what has already happened may be used for forward-looking decision-making.

— 2020 Scrum Guide page 7

## The Importance of Empiricism

Empiricism insists that experience and evidence should form the basis of belief and knowledge, not just graphs and metrics.

For example, a chart might say a team is completing 20 backlog items a week. Does that mean you'll complete 20 backlog items next week?

You might *empirically* know that two developers will be on vacation next week, or perhaps one developer is coming down with the flu.

Maybe World Cup starts next week and your entire offshore team is in Brazil?

The trend on the chart may indicate that the team will complete 20 backlog items next week, but you empirically know that sickness or vacation or the start of the World Cup will likely decimate that number.

That's an example of empirical knowledge being a much better and a much more informed indicator of future performance than a burn-down chart.

## Who Can Cancel a Sprint?

The Scrum Certification exam will undoubtedly ask you who has the power to cancel a Sprint, and under what conditions the Sprint can be canceled.

A Sprint could be canceled if the Sprint Goal becomes obsolete.

Only the Product Owner has the authority to cancel the Sprint.

— 2020 Scrum Guide page 7

These two points surprise people for the following three reasons:

1. It is indeed possible for an active Sprint to be canceled
2. The only reason to cancel a Sprint is if the Sprint Goal becomes obsolete
3. Only the Product Owner can cancel a Sprint, not the Scrum Master or any stakeholder

Just by the authoritative nature of the name, people think a Scrum Master would have the authority to cancel a Sprint.

Scrum Masters don't. Scrum Masters don't really have much authority in Scrum.

A Scrum Master can't cancel a Sprint. Only the Product Owner can, and there is only one reason why a Sprint can be canceled: the Sprint Goal has become obsolete.

That means none of the following reasons are justifications to cancel the Sprint:

- The development team needs more time to finish their work
- The lead developer quit and change the GitHub password
- The building flooded and there's no power
- The Scrum Master has gone on vacation
- The CEO wants a different feature developed

There are plenty of reasons why someone in the organization might want to cancel a Sprint, but the only reason a Sprint can be canceled is if the Sprint Goal has become obsolete, and the only person allowed to make that call is the Product Owner.

## What isn't said about the Sprint

People want the Scrum Guide to answer all of their questions about how to run a big project. Scrum doesn't do that.

Scrum provides a set of basic rules written out in a guide that's less than 4000 words long. I wrote college papers longer than that.

If Scrum doesn't provide a rule or guideline around something, then it's up to the group of self-managing, self-organized, highly motivated professionals on the Scrum Team to figure it out on their own.

## When to Start the Sprint?

The Scrum Certification exam often tries to trick you into applying some arbitrary rule that seems to makes logical sense but doesn't actually appear anywhere in the Scrum Guide.

For example, say a product has two Scrum Teams working on it, which happens quite regularly.

- Should those two teams start their Sprints on the same day?
- Should those two teams end their Sprints on the same day?
- Should those two teams have Sprints of equal length?
- Should those two teams have the same Scrum Master?

On the surface, all of those assertions seem reasonable. It seems like a nice way to keep the two Scrum Teams in sync.

At the same time, that approach might create chaos. Imagine stakeholders having to go to two Sprint Reviews on the same day. That might be too much for them.

Or maybe two teams who finish their Sprints on the same day would end up pushing their releases into production on the same day, which might cause panic for system administrators?

So maybe it'd be better to stagger the Sprints? Have one team start a 4 week Sprint on the first of the month, and have the other start the Sprint in the middle of the month?

## No Rule Means There's No Rule

The fact is, the Scrum Guide doesn't care what those two teams do. The teams are expected to figure it out on their own.

If the Scrum Guide doesn't speak directly to the topic, then there are no rules about it.

The Scrum Guide **does** say two teams working on the same project must share the same Product Backlog and the same Product Owner.

- Should their Sprints be the same length?
- Should their Sprints start on the same day?

Scrum doesn't provide any guidance on these topics. It's assumed your team will use empirical measures and apply lean thinking and come up with a plan that is best for them.

Scrum wants you to figure out what's best for you. It just wants you to do that within the confines of the incomplete framework it provides.

## Non-Empirical Scrum Metrics

Here are some definitions of the three types of charts mentioned in the Scrum Guide.

You won't be tested heavily on these on the exam, but you may get a question or two asking at a very general level what each of these charts do.

## Burn-Down Chart

The burn-down chart is a graphical representation of the amount of work remaining in a project versus time.

The burn-down chart tracks the progress of a project by showing the remaining work that needs to be completed on the vertical axis and the time on the horizontal axis. The burn-down chart starts with the total amount of work to be completed at the beginning of the project, and as work is completed, the line on the chart gradually moves down until it reaches zero at the end of the project.

Burn-down charts are useful because they help teams track progress over time and adjust their approach as necessary. For example, if the team is falling behind schedule, they can identify this early on and make changes to get back on track.

## Burn-Up Chart

The burn-up chart is similar to a burn-down chart but shows progress differently. Rather than showing the remaining work on the vertical axis, a burn-up chart shows the amount of work completed over time. The chart starts with zero completed work at the beginning of the project, and as work is completed, the line on the chart moves up until it reaches the total amount of work to be completed at the end of the project.

Burn-up charts are useful because they show progress towards a goal and can help teams identify whether they are on track to complete the project on time.

## Cumulative Flow Chart

The cumulative flow chart is a graphical representation of the flow of work in a project. It shows the amount of work in progress at any given time, as well as the rate at which work is being completed. The chart has a horizontal axis representing time and a vertical axis representing the number of tasks.

The chart starts with a backlog of tasks at the beginning of the project and shows how tasks move through different stages of completion over time. For example, a task might start in the "to do" column, move to "in progress", and then finally move to "completed".

Cumulative flow charts are useful because they help teams identify bottlenecks in their workflow and optimize their processes to increase efficiency. By tracking the rate of completion over time, teams can also identify whether they are on track to complete the project on time.

## Chapter 10: Sprint Planning

Scrum is simple. It really is.

I often hear people say they hate Scrum because it's too complicated, but I just don't see how that's possible. At the heart of it, Scrum just says this:

- Teams should take a bit of time every month or so to plan out the work they're going to do
- Developers should meet for 15 minutes every workday just to get the ball rolling again
- At least once a month developers should share what they've done with stakeholders
- After talking to stakeholders, the team should talk a bit about what did and didn't work

That's it. That's the basics of Scrum right there.

I honestly don't know how anyone could say a strategy like that is a bad thing. It seems pretty basic, pretty reasonable, and pretty straightforward to me, and it all starts with Sprint Planning.



Figure 22. Sprint Planning is the first of the four Scrum events that take place within a Sprint.

Sprint Planning initiates the Sprint by laying out the work to be performed for the Sprint.

This resulting plan is created by the collaborative work of the entire Scrum Team.

— 2020 Scrum Guide page 8

Harking back to a theme we've crowed about before: Sprint Planning happens within the Sprint, not before the Sprint begins.

Many questions on the Scrum Master certification exam will suggest that Sprint Planning happens before the Sprint, but that's always a red herring. None of the Scrum events happen *before* or *after* a Sprint. Every Scrum event takes place *within* the Sprint.

## Product Backlog Items and the Product Goal

The Product Owner ensures that attendees are prepared to discuss the most important Product Backlog items and how they map to the Product Goal.

— 2020 Scrum Guide page 8

We learned earlier in the section about the Product Owner that the PO is the one accountable for maximizing the value of the product created by the Scrum Developers.

Here the Scrum Guide re-emphasizes that point, stating that the Product Owner comes to the Sprint Planning meeting with a list of the most important features for the team to work on.

## Who's Invited to the Sprint Planning Meeting?

The Scrum Team may also invite other people to attend Sprint Planning to provide advice.

— 2020 Scrum Guide page 8

Notice how anyone can be invited to participate in the Sprint Planning meeting. That's a significant point and a concept you'll be tested on often.

There are often multiple Scrum Teams working on a single product.

Remember, the maximum size of a Scrum team is 10 people, but it's not unusual for an enterprise software development project to have thirty or forty developers working on it. That means there will be at least four or five Scrum teams all working on the same product at the same time.

## Minimizing Inter-team Dependencies

So how do multiple teams avoid stepping on each other's toes, working on the same backlog items at the same time, or working on items that might block another team's progress?

How do teams avoid being blocked by a dependency their chosen work has on something that has to be completed by another team first?

One way to manage these issues is to have representatives from other Scrum Teams attend your team's Sprint Planning meeting. Representatives from other teams can share their progress, insights, and even deliverable dates for dependencies your project might have on their work.

Or better yet, if you know that certain Product Backlog items have dependencies on work that might not get completed by another team by the time you need them, then you can just pick different backlog items to work on.

The fewer the dependencies your work has on other teams, the better.

## Test Yourself

The Scrum Developers might want to invite someone outside the Scrum Team to attend the Sprint Planning meeting to discuss pertinent issues.

Which one of the following is true about people not on the Scrum Team and Sprint Planning?

- A) Outsiders can discuss how Product Backlog items align with corporate strategy and business goals
- B) Outsiders can discuss how certain Product Backlog items may be of importance to customers
- C) Outsiders can discuss dependencies on other items other teams are developing
- D) Outsiders can discuss problems that might be encountered in the current Sprint
- E) All of the above are true.

Option E is correct.

There is nothing in the Scrum Guide that forbids Scrum Developers from inviting people outside the team to the Sprint Planning meeting, and there are no pragmatic restriction on what outsiders can talk about.

If someone outside the Scrum Team can provide guidance on how to best plan the Sprint, they are more than welcome to speak up and contribute during Sprint Planning.

## Test Yourself

It is the Scrum Master's responsibility to ensure all Sprint Planning attendees are ready and prepared to discuss the Product Backlog.

- True
- False

This is false. It's the *Product Owner* who ensures attendees are prepared to discuss how Product Backlog items map to the Product Goal.

Don't over-assign responsibilities to the Scrum Master. The name 'Scrum Master' is deceptively authoritative. The reality is, the Scrum Master has very few powers on the Scrum Team outside of coaching people on how to implement Scrum effectively.

The Product Owner owns the Product Backlog. Duties related to the Product Backlog typically fall in the Product Owner's lap.

## Answering Why, What, and How

The goal of Sprint Planning is to come up with good answers to the questions of why, what, and how. You'll be asked 'straight up' on the Scrum Master certification exam which three questions the Sprint Planning event tries to solve.

Sprint Planning addresses the following topics:

- Topic One: Why is this Sprint valuable?
- Topic Two: What can be Done in this Sprint?
- Topic Three: How will the chosen work get done?

Sprint Planning is timeboxed to a maximum of eight hours for a one-month Sprint. For shorter Sprints, the event is usually shorter.

— 2020 Scrum Guide page 8

## Timeboxing Sprint Planning to 8 Hours

To pass the Scrum certification exam you have to know the timeboxes for Scrum events like you know the back of your hand. Memorize these:

- Sprint Planning is timeboxed to a maximum of 8 hours for a one-month Sprint
- The Daily Scrum is timeboxed to a maximum of 15 minutes
- The Sprint Review is timeboxed to a maximum of 4 hours
- The Sprint Retrospective is timeboxed to a maximum of 3 hours

## Test Yourself

Which of the following Scrum events can last the longest?

- A) Daily Scrum
- B) Sprint Review
- C) Sprint Planning
- D) Sprint Retrospective
- E) The Review, Planning, and Retrospective are all timeboxed to 4 hours
- F) There is no time limit for the Sprint Planning

The answer to this question is C, Sprint Planning.

Sprint Planning is timeboxed to a maximum of 8 hours. Hopefully, your team can get it done a bit faster.

## Why is the Sprint valuable?

The Product Owner proposes how the product could increase its value and utility in the current Sprint.

— 2020 Scrum Guide page 8

It is the Product Owner who is responsible for ensuring the work of the Scrum Team produces the greatest amount of value.

The Product Owner knows what needs to be built to make the product better. However, the Product Owner has no innate knowledge of *how* to build it. That's the job of the developers.

## Negotiating Product Backlog Item Selection

During Sprint Planning, the Product Owner can only propose their ideas of what should be built next.

The Developers may have pragmatic objections to what the Product Owner proposes and push for other Product Backlog items to be made part of the current Sprint instead.

Just think about a Scrum Team constructing a house.

The Product Owner would likely want the kitchens and the bathrooms done first, but the foundation of the house may not be laid. In that case, the developers would need to explain how the kitchen and the bathrooms will need to wait for a future Sprint as the home's foundation has to be poured first.

Figuring out exactly what's required to implement a Product Backlog item is known as *decomposition*.



*Figure 23. The Developers will negotiate with the Product Owner over which PBIs can feasibly be selected for the Sprint.*

## Test Yourself

Who on the Scrum Team is responsible for maximizing the value of the work performed by the developers? (Choose 1)

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

Option B is correct.

Maximizing the value of the work performed by the Scrum Team is the job of the Product Owner.

## Test Yourself

Who first proposes an initial plan for the Sprint at the Sprint Planning meeting? (Choose 1)

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

Option B is correct.

It is the Product Owner who presents a list of Product Backlog items they would like to have built in the current Sprint.

This becomes the starting point for negotiations, suggestions, decomposition and compromises during Sprint Planning.

### NOTE

The term *epic* is used by some Agile practitioners to describe large Product Backlog items that will take multiple Sprints to complete. While most people in the Agile community are familiar with the term, it is not a term that is ever used in the Scrum Guide.

## Sprint Planning and the Sprint Goal

The Sprint Goal must be finalized by the end of the Sprint Planning meeting, and while other things may change during a Sprint, the Sprint Goal is one of the Scrum artifacts that is not allowed to be edited, adjusted, or changed once Sprint Planning has concluded.

The whole Scrum Team then collaborates to define a Sprint Goal that communicates why the Sprint is valuable to stakeholders.

The Sprint Goal must be finalized before the end of Sprint Planning.

— 2020 Scrum Guide page 8

Each Sprint needs a goal. The Sprint Goal keeps the developers focused throughout the Sprint.

The Sprint Goal also provides another important function - it provides transparency into the Sprint, as it allows stakeholders to know what developers are working towards.

The Sprint Goal must be finalized before the Sprint Planning meeting ends and it cannot change throughout the Sprint. The Sprint Plan can change, and the items in the Sprint Backlog can change. It's expected that those things will change as conditions change throughout the Sprint. But the Sprint Goal must be finalized before the Sprint Planning meeting ends, and it cannot change during the Sprint.

### Test Yourself

What happens if it becomes clear towards the end of the Sprint that the team will not achieve the Sprint Goal? (Choose 1)

- A) The Sprint is canceled and a new Sprint Planning meeting takes place
- B) The next Sprint adopts the current Sprint's Goal continuously until the goal is achieved
- C) The developers update the Sprint Goal so that it is achievable by the end of the Sprint
- D) The developers talk about the Sprint Goal during the Sprint Retrospective

Option D is correct.

It's not unusual for a Sprint Goal to go unfulfilled. Sometimes things just don't go according to plan.

If the Sprint Goal is not achieved, the Scrum Team talks about what they can do better during the Sprint Retrospective.

Nothing ever gets automatically rolled over from one Sprint into the next. Each Sprint starts new with an empty Sprint Backlog, as it's assumed that since conditions, expectations, and realities are constantly changing, what made sense when the prior Sprint was planned won't necessarily make sense for the current one.

The only exception to this rule is when suggestions on how to improve team practices made during the Sprint Retrospective get added to the next Sprint's Sprint Backlog. Other than that one corner case, items in the Sprint Backlog are never carried over from one Sprint to the next.

## Test Yourself

Who creates the Sprint Goal? (Choose 1)

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

Option D is correct.

The Scrum Team as a whole creates the Sprint Goal.

## Product Backlog Item Selection

Through discussion with the Product Owner, the Developers select items from the Product Backlog to include in the current Sprint.

The Scrum Team may refine these items during this process, which increases understanding and confidence.

— 2020 Scrum Guide page 8

During Sprint Planning, the Product Owner proposes what they believe should be built during the Sprint. The Product Owner explains what they believe will provide the most value to stakeholders and customers. However, it's the developers who decide which Product Backlog items get added to the Sprint.

Going back to the home construction analogy, the construction workers know more about how to build a house than the homeowner.

The developers know what dependencies exist, what order certain Product Backlog items need to be created in, and what's the best path forward in order to complete the highest value Product Backlog items.

The developers, not the Scrum Master or the Product Owner, have the final say over what gets added to the Sprint.

## Product Backlog Refinement

While Sprint Planning is an opportunity for the developers to talk about Product Backlog items, clarify them and refine them, this is not the only time developers are allowed to talk with the Product Owner. The developers can call up the Product Owner at any time during the Sprint to clarify details about Product Backlog items.

Quite often the Product Owner works out of the same war room as the developers. That way the PO can answer developer's questions about the product as soon as they arise.

The Scrum Master certification exam will often provide an incorrect option that indicates there are only certain times a developer can talk to a stakeholder or Product Owner. Discussion between everyone on the Scrum Team and every stakeholder in the organization is never discouraged in the Scrum Guide. The more conversations the better!

### Test Yourself

The Product Owner selects the Product Backlog items developers will work on during the Sprint.

- True
- False

This is false.

The Product Owner can prioritize the Product Backlog and inform the developers about which Product Backlog items provide the most value, but the developers have the final say over which items they build during the Sprint.

### Test Yourself

Sprint Planning is the only time Scrum Developers are allowed to talk to the Product Owner to discuss Product Backlog items.

- True
- False

This is false.

The Product Owner should always be available to answer questions about the product. Conversations between the Product Owner, the developers, the Scrum Master, and the stakeholders should happen freely and openly. Nothing in the Scrum Guide forbids it.

## What can be Done in this Sprint?

Selecting how much can be completed within a Sprint may be challenging. However, the more the Developers know about:

- their past performance,
- their upcoming capacity, and;
- their Definition of Done,

the more confident they will be in their Sprint forecasts.

— 2020 Scrum Guide page 8

According to the Scrum Guide, it's the Scrum Developers who are expected to estimate how much can be done in a Sprint. It's up to the Scrum Developers to know about their capacity, past performance, and ability to follow through on the Definition of Done to estimate how much work they can accomplish.

This makes sense. After all, it is the developers who choose the PBIs they plan to implement in the Sprint. If they were unable to estimate their work capacity, they wouldn't know how many Product Backlog items to choose.

### Test Yourself

Who estimates how much work can be accomplished in a Sprint? (Choose 1)

- A) The Scrum Master
- B) The Product Owner
- C) The Scrum Developers
- D) The Scrum Team as a whole

Option C is correct.

The Developers are the ones who select how many Product Backlog items to include in the Sprint. The Developers are the ones who are estimating how much they believe can be accomplished.

## How will the chosen work get done?

For each selected Product Backlog item, the Developers plan the work necessary to create an Increment that meets the Definition of Done.

This is often done by decomposing Product Backlog items into smaller work items of one day or less. How this is done is at the sole discretion of the Developers. No one else tells them how to turn Product Backlog items into Increments of value.

— 2020 Scrum Guide page 8

## Product Backlog Decomposition

A Product Backlog item may take months to complete.

In Scrum, the Developers need to break epic Product Backlog items down into smaller pieces. This is known as decomposition.

The goal of decomposition is to break each Product Backlog item down into a series of chunks that can be estimated to take a day or less to complete.

## No Points or User Stories in Scrum

Note that the Scrum Guide never talks about *points* or *user stories*. It never uses the term *epic* either, which is a term commonly thrown around by Agile teams to describe a really big Product Backlog item that might take multiple Sprints to complete.

Quite often you will see a question on the Scrum Master certification exam where one of the answers talks about breaking user stories down into a certain number of points. Those answers are always wrong.

That's not to say Scrum Teams can't use points or user stories if they find them helpful. It's just that they're not discussed in the Scrum Guide and are not a part of the Scrum framework.

The Scrum Master certification exam is testing you on Scrum, not peripheral processes and methodologies people sometimes integrate into Scrum.

## Test Yourself

Which one is the most 'Scrum way' to approach a complicated Product Backlog item?

- A) Have the development team break the Product Backlog item down into smaller user stories.
- B) Have the Product Owner break the Product Backlog item down into smaller user stories.
- C) Break the Product Backlog item down into 1-point increments, representing 1 day of work.
- D) Try to break the Product Backlog item into multiple work items of one day or less

Option D is correct.

The term 'user story' is never mentioned once in the Scrum Guide. Neither is 'points.' That makes the first three options wrong.

Given an epic Product Backlog item, the Scrum Developers should try to decompose it into multiple, smaller work items that will take a day or less for a developer to complete.

## The Outcome of Sprint Planning

The Sprint Goal, the Product Backlog items selected for the Sprint, plus the plan for delivering them are together referred to as the Sprint Backlog.

— 2020 Scrum Guide page 9

Note that the Sprint Backlog is more than just the set of Product Backlog items the team has selected for the Sprint. It also includes a Sprint Goal that cannot change during the Sprint, along with a plan that is expected to change daily.

### Test Yourself

The Sprint Backlog is composed of: (Choose 1)

- A) The Product Backlog items selected for the Sprint
- B) The Product Backlog items selected for the Sprint, and the Sprint Goal
- C) The Product Backlog items selected for the Sprint, the Sprint Goal, and the Sprint Plan
- D) The Product Backlog items selected for the Sprint, the Sprint Goal the Sprint Plan, and the Product Goal

Option C is correct.

The Sprint Backlog consists of the Product Backlog items selected for the Sprint, the Sprint Goal and the Sprint Plan.

### Test Yourself

Who is allowed to view the Sprint Goal? (Choose 1)

- A) The Scrum Master
- B) The Scrum Developers
- C) Only members of the Scrum Team
- D) The Scrum Team and stakeholders

Option D is correct.

The Sprint Goal helps to build transparency into the development process by allowing stakeholders to know what the team is trying to achieve during the current Sprint.

## Test Yourself

Which of the following are the most *empirical* measures of how much the Scrum Developers can accomplish in a given Sprint? (Choose 2)

- A) Burndown charts
- B) Past performance
- C) Burnup charts
- D) Upcoming capacity

Options B and D are correct.

Upcoming capacity and knowledge of past performance are more empirical measures than burndown or burnup charts.



## Chapter 11: The Daily Scrum

Some frameworks and methodologies have a *Daily Standup*.

We don't have *Daily Standups* in Scrum.

First of all, the term is ableist. Not everyone can stand.

Secondly, the idea behind a *Daily Standup* is that it won't last long because people will eventually get tired of standing.

Scrum practitioners don't believe the infliction of pain is the proper way to motivate people.

The Daily Scrum is a 15-minute event for the Developers of the Scrum Team.

The purpose of the Daily Scrum is to inspect progress toward the Sprint Goal and adapt the Sprint Backlog as necessary, adjusting the upcoming planned work.

— 2020 Scrum Guide page 9



Figure 24. The purpose of the Daily Scrum is for the developers to metaphorically get the ball moving again.

## Adapting to Change

Things change from day to day, and changes can mess up your plans.

That's why developers get together everyday for a Daily Scrum so they can talk about anything that is impeding or accelerating their progress towards the Sprint Goal.

If things are going sideways, the developers adapt. If things are going better than expected, the developers adapt.

The goal of the Daily Scrum is for developers to talk, discuss progress, and quickly solve any problems that have arisen since they last spoke.

The Daily Scrum is timeboxed to 15 minutes.

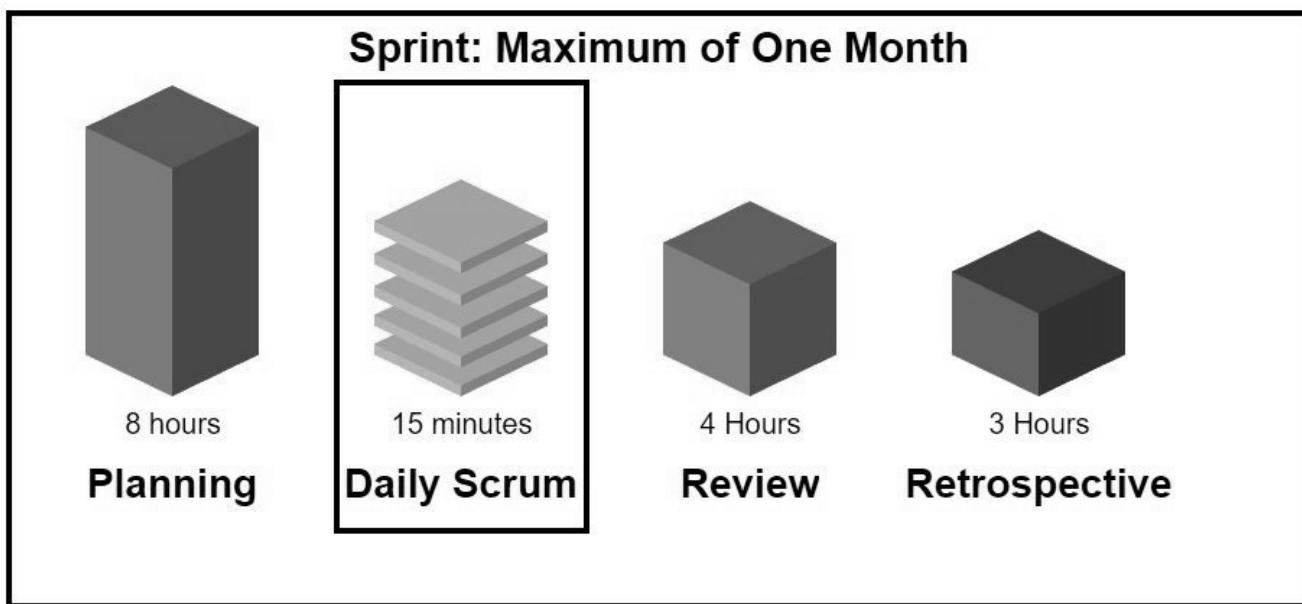


Figure 25. The Daily Scrum takes place on every workday during the Sprint.

## Keep the Time and Location Consistent

To reduce complexity, it is held at the same time and place every working day of the Sprint.

— 2020 Scrum Guide page 9

All Scrum events are supposed to be held at the same time and in the same location. This applies to the Daily Scrum as well.

This particular point is more salient when developers work in the same office building. You don't want the developers wasting 10 minutes every day trying to figure out which conference room to go to.

In the post-COVID age where teams are largely remote, 'the same time and location' now equates to a Zoom or Microsoft Teams call that takes place at the same time every day.

## Demystifying the Daily Scrum

Let's get real about the Daily Scrum for a minute.

People always complain that Scrum has too many time wasting meetings. When they say that, they're usually talking about the Daily Scrum.

If your Daily Scrum feels like a waste of time, you're doing the Daily Scrum wrong. The Daily Scrum shouldn't feel like a formal, highly structured meeting. The Daily Scrum should feel casual and natural.

Scrum doesn't try to put an overwhelming amount of structure and ceremony around the development process. At its heart, Scrum just says:

- Plan your work before you start. That's why we have the Sprint Planning meeting.
- Regularly review your work with stakeholders. That's why we have the Sprint Review.
- Regularly talk about how you can improve things. That's why we have the Sprint Retrospective.

It's hard to think someone could argue that any of these meeting are bad. These activities just naturally evolve from any team that's serious about product development.

Scrum also says that every workday the developers should talk to each other about what they're doing. That hardly seems unreasonable.

## Keep the Daily Scrum Simple

Scrum isn't some framework that demands developers use some crazy set of new and revolutionary techniques. All Scrum really does is document the practices good development teams already do.

Did you ever work on a group project in High School or College? One where you had to meet-up with five friends in the library a few times to get the project finished?

What did you all do for the first 15 minutes when you got together?

- You bitched about your prof and gossiped about the other teams
- You talked about what you did and didn't get done
- You asked others for help with things you couldn't manage
- You talked about things that were blocking your progress and you offered help to others

You'd have a pretty dysfunctional team if y'all didn't chat casually about these types of things when you got together.

The Daily Scrum is simply trying to capture the natural energy and enthusiasm highly motivated teams should already have when working collaboratively on a project.

## Rules of the Daily Scrum

The basic rules of the Daily Scrum are as follows:

- The Daily Scrum should take place at the same time every workday
- The Daily Scrum should take place in the same location every day
- During the Daily Scrum developers talk about their work, not last night's hockey game
- The Daily Scrum should never last longer than 15 minutes

## Daily Scrum Time and Location

When I think about developers chatting about their work, their progress and their blockers, I envision them doing it when they all get into the office in the morning.

However, the Daily Scrum doesn't have to take place in the morning. It can take place at lunch-hour or even at the end of the day.

In fact, with modern teams distributed across the globe, what might be the start of the day for some members of the team might be the afternoon for others.

The Scrum Guide doesn't say what time the Daily Scrum needs to take place, only that it should take place every workday - not every day, just every *workday*.



Figure 26. The Daily Scrum is for the developers and managed by the developers. If the Scrum Master or the Product Owner attends, they do so at the behest of the developers.

## Test Yourself

Which one of the following is the responsibility of the Scrum Master during the Daily Scrum?

- A) The Scrum Master should coach the developers about keeping the Daily Scrum timeboxed to 15 minutes
- B) The Scrum Master should ensure each developer gets a chance to speak
- C) The Scrum Master starts the Daily Scrum by asking the 3 standard Daily Scrum questions
- D) The Scrum Master starts the Daily Scrum by asking every developer to stand up.

The only correct option here is A.

The only responsibility the Scrum Master has with regards to the Daily Scrum is ensuring it is run in such a way that complies with the rules of Scrum. In regards to this quiz question, that means the only correct option is keeping the Daily Scrum to less than 15 minutes.

Past Scrum Guides mentioned the 3 Daily Scrum questions:

- What did you do yesterday?
- What do you plan to do today?
- Is anything impeding your progress?

These questions were removed from the 2020 Scrum Guide. They can be asked if the developers find them helpful, but they are certainly not a requirement of the Daily Scrum.

## Test Yourself

The development team has decided to only work one day a week and will only hold the Daily Scrum on Mondays.

How do you react as a Scrum Master? (Choose 1)

- A) Inform the team Scrum development cannot happen only one day a week
- B) Have Human Resources talk to the development lead about their proposed work schedule
- C) Cancel the Sprint and consult the Product Owner
- D) Inform the team this is fine so long as the only workday is Monday

Option D is correct.

There's nothing in the Scrum Guide that forbids a team from working only one day a week.

The only Scrum requirement with regards to this question is that the Daily Scrum happens every *workday*. So if the team holds the Daily Scrum on the Monday, then the work should be happening on the Monday too.

The Daily Scrum is a 15-minute event for the Developers of the Scrum Team. To reduce complexity, it is held at the same time and place *every working day* of the Sprint.

— 2020 Scrum Guide page 9

## Test Yourself

The Scrum team wants to move the Daily Scrum to the atrium on Fridays to help them wind down after a busy workweek. How do you, as a Scrum Master, respond? (Choose 1)

- A) Explain to them why the Daily Scrum always occurs in the same place
- B) Allow the developers to be self-managed and hold their Daily Scrum in the Atrium on Fridays
- C) Confirm with the Product Owner that it is okay to change the Daily Scrum location on Fridays
- D) As the Scrum Master, work to remove any impediments standing in the way of running the Friday Scrum in the atrium

Option A is correct.

The Scrum Guide says the Daily Scrum should always take place at the same time and place.

This is true for all Scrum events. For the sake of consistency, their time and location should remain constant.

## When Scrum Masters and POs do Development

Have you ever seen a Scrum Master or Product Owner do some work that becomes part of a usable Increment?

When they do, they become developers themselves.

If the Product Owner or Scrum Master is actively working on items in the Sprint Backlog, they participate as Developers.

— 2020 Scrum Guide page 9

This is an interesting statement that answers several questions about how Scrum works.

People often wonder if a Scrum Master or Product Owner is allowed to actively do development. The answer is a big YES.

If a Scrum Team of five people is cast away on a deserted island and scrambling to build a shelter before a storm comes, everyone on that Scrum Team is going to pitch in. If your life is on the line, you're not going to refuse to help build a shelter just because you've accepted the designation of *Scrum Master*.

## The Blended Role Anti-Pattern

On small teams and startups, the Scrum Master might also be the Product Owner and they might be a developer as well. It might not be a best practice, but there's nothing that forbids it. More to the point, it might make a lot of sense in a really small development firm.

So yes, a developer can also be a Scrum Master or a Product Owner or vice versa. This is often referred to as a *blended role*, and since it's not optimal, many in the Scrum community consider it to be an anti-pattern. However, anti-pattern or not, there is nothing in the Scrum Guide that forbids it, and for Scrum teams in small companies, it is often a reality.

The only requirement is that if a Scrum Master or Product Owner does development, they are expected to attend the Daily Scrum and participate as though they were a developer, not the Scrum Master or PO. They relinquish their Scrum Master or Product Owner accountabilities while the Daily Scrum takes place.

## Who Participates in the Daily Scrum?

The Developers can select whatever structure and techniques they want, as long as their Daily Scrum focuses on progress toward the Sprint Goal and produces an actionable plan for the next day of work. This creates focus and improves self-management.

— 2020 Scrum Guide page 9

The Daily Scrum is for the developers. It should be run by the developers, organized by developers and managed by the developers. How they manage or organize it is up to them.

Anyone can *attend* the daily Scrum. If the developers want to hire a circus clown to create balloon animals while the Daily Scrum proceeds, then all the power to them. But only the developers are supposed to *participate*.

Now that's not to say the developers can't ask the Scrum Master or Product Owner a question during the Daily Scrum. That may be necessary to properly adapt their plan towards the Sprint Goal. But the Scrum Master, Product Owner and stakeholders shouldn't be active participants driving the event. The Daily Scrum must be driven by the developers.

## Purpose of the Daily Scrum

Daily Scrums improve communications, identify impediments, promote quick decision-making, and consequently eliminate the need for other meetings.

— 2020 Scrum Guide page 9

Things change from day to day. The Daily Scrum is a time for developers to deal with issues that have arisen that may delay their progress and put the Sprint Goal in jeopardy.

Hopefully, having the whole team of developers together in the Daily Scrum will help to bring about quick solutions to problems that may have arisen.

The Daily Scrum is not the only time Developers are allowed to adjust their plans.

They often meet throughout the day for more detailed discussions about adapting or re-planning the rest of Sprint's work.

— 2020 Scrum Guide page 9

Don't ever think that the Daily Scrum is the only time developers are allowed to talk about their work, or that it's the only time to change the Sprint Plan.

If a team of construction workers was putting up a roof, and a wind gust blew all their shingles away, would they wait until the next day's Daily Scrum to form a new plan, or would they reformulate their plans right away?

Developers can meet with each other any time they like. They can schedule additional meetings as a group. They can meet one-on-one at each other's desks. They can eat dinner together.

There's nothing in the Scrum Guide that restricts communication between developers, stakeholders, Product Owners, Scrum Masters, or anyone else. The only thing the Scrum Guide recommends is a few timeboxed events that guarantee opportunities for communication, transparency, and adaptation to take place.

## Test Yourself

A critical bug has appeared in your code that may put the Sprint Goal at risk. What should you, as a developer, do? (Choose 1)

- A) Speak to the developers as soon as possible to find a way to adapt the Sprint plan
- B) Bring the issue up in the next day's Daily Scrum meeting
- C) Inform the Scrum Master and have the Scrum Master remove the impediment
- D) Pass the issue to the Quality Assurance team and continue working on Product Backlog items

Option A is correct.

Any time an important issue comes up it should be addressed as soon as possible. If a critical bug appears in your code, and you think it will impact the Sprint Goal, then meet with your fellow developers right away and see if you can adapt by adjusting the Sprint plan.

Don't ever let the Scrum Guide and the various Scrum events and artifacts impede pragmatic thinking. If a problem arises that needs to be taken care of immediately, take care of it immediately. Don't wait for the next Scrum event to adapt.

## Does Scrum Really Have Too Many Meetings?

When people say Scrum has too many meetings, I ask them which ones they think should be removed?

- Should there be no planning?
- Should there be no product reviews with clients?
- Should there be no retrospective to reflect on the Sprint?

Or is the Daily Scrum the problem? Do they think having developers talking to each other in the Daily Scrum for a few minutes every day is too much?

If a team of six or seven developers is working on a project that burns through \$250,000 worth of budget money a month, I'd say it was a problem if those developers didn't talk to each other for a few minutes every day.

Scrum doesn't have too many meetings. Scrum gets it just about right.



## Chapter 12: Sprint Review

On any project, not just in the domain of Scrum, stakeholders should continually get updates on the state of the product they've invested in.

That's one of the ideas behind the continuous delivery of software - to constantly deliver software to your clients and stakeholders every time a new feature or Increment of value has been completed.

The more often the stakeholders see new features added to the product, the more in tune they will be with the pace at which the project is progressing. Regular updates allow stakeholders to inspect what has been completed and provide feedback so the team can adapt accordingly.

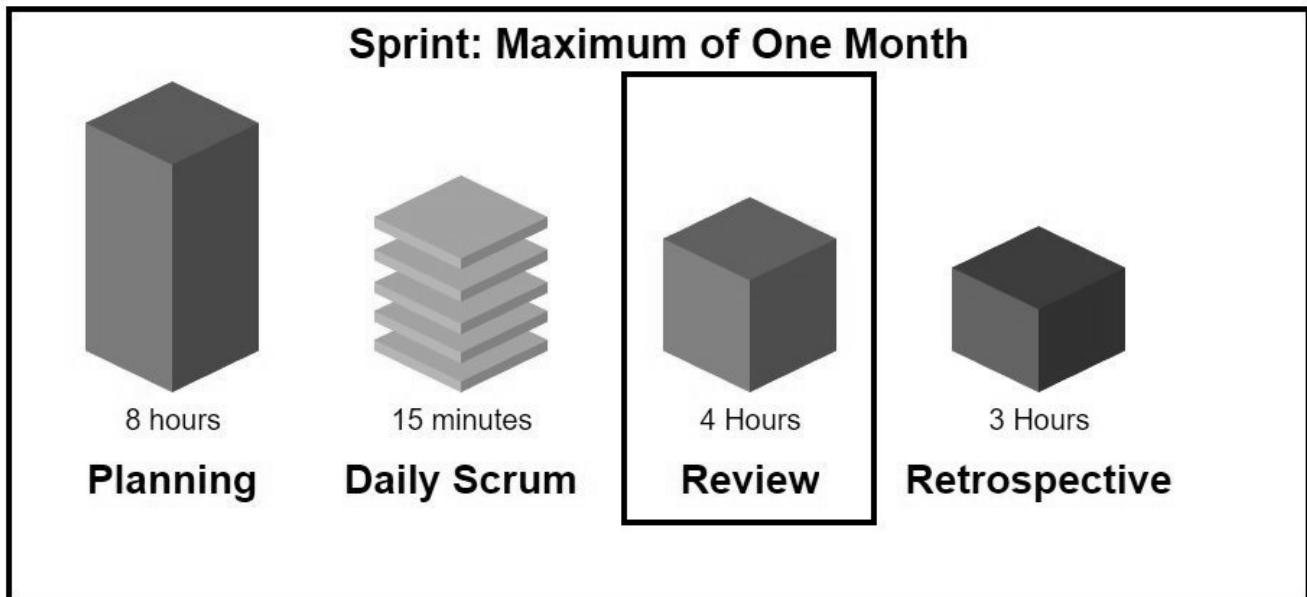


Figure 27. The Sprint Review is the second-last Scrum event to take place within the Sprint.

The Sprint Review exists to make sure stakeholders get at least one chance to inspect the work completed during every Sprint.

The purpose of the Sprint Review is to inspect the outcome of the Sprint and determine future adaptations.

The Scrum Team presents the results of their work to key stakeholders and progress toward the Product Goal is discussed.

— 2020 Scrum Guide page 9

The Sprint Review is the one Scrum event where everyone is invited and allowed to participate. That not only includes everyone on the Scrum Team but stakeholders as well.

The purpose of the Sprint Review is pretty self-explanatory. It's for everyone to review what was done and talk about what they like, what they don't like, and where they think things should go next.

## Sprint Review Participants

Everyone gets to speak at the Sprint Review. Developers can speak to stakeholders, stakeholders can speak to the Scrum Master, and the Product Owner can speak to developers.

There are no restrictions in the Scrum Guide about developers ever talking to stakeholders, the Scrum Master talking to stakeholders, or the Product Owner talking to stakeholders.

The Sprint Review intentionally and concertedly brings everyone together once per Sprint, but there's no rule that says developers can't talk to stakeholders about development when they pass each other in the hallways at work.

The more openness and transparency the better, as far as Scrum is concerned.

### Test Yourself

Which of the following is true about the Sprint Review? (Choose 1)

- A) Only the Scrum Master is allowed to talk directly to stakeholders
- B) Only the Product Owner is allowed to talk directly to stakeholders
- C) Developers are not allowed to talk directly to stakeholders
- D) The Sprint Review is the only Scrum event where both stakeholders and the Scrum Team are invited

Option D is correct.

Everyone who has a dog in the hunt is invited to the Sprint Review.

As for the other options, there are no rules anywhere in the Scrum Guide that forbids any members of the Scrum Team from talking to the stakeholders.

### What to show in the Review?

There's a small detail left out of this discussion of the Sprint Review which is somewhat important. For example, the following sentence doesn't exactly specify what the stakeholders get to see during the Sprint Review.

During the event, the Scrum Team and stakeholders review *what was accomplished* in the Sprint and what has changed in their environment.

— 2020 Scrum Guide page 9

The term **what was accomplished** is somewhat vague.

## Only Completed Increments in the Sprint Review

We discover in the Scrum Guide's section on artifacts that only *completed Increments*, that is work items that have met the Definition of Done, are to be presented at the Sprint Review.

Multiple Increments may be created within a Sprint. The sum of the Increments is presented at the Sprint Review thus supporting empiricism.

Work cannot be considered part of an Increment unless it meets the Definition of Done.

— 2020 Scrum Guide page 12

Any work that is partially done is not to be presented. Only Increments, or the sum of all the Increments created during the Sprint, are to be discussed at the Sprint Review.

## A Discussion on What to do Next

Based on this information, attendees collaborate on what to do next.

The Product Backlog may also be adjusted to meet new opportunities.

— 2020 Scrum Guide page 9

The Sprint Review gives everyone on the team, and the stakeholders, a chance to talk about what they like, what they don't like, and what they think should change in terms of how the product is being developed.

If the stakeholders don't like the direction development is taking and want some proposed features removed, the Product Owner may remove those features from the Product Backlog.

If there are features the stakeholders decide they want, then the Product Owner can either add them to the Product Backlog or prioritize them if they already exist.

## Another Scrum Pillars Feedback Loop

The Sprint Review helps create transparency, is an opportunity for inspection, and allows for adaptation.

The Sprint Review is the second to last event of the Sprint and is timeboxed to a maximum of four hours for a one-month Sprint.

For shorter Sprints, the event is usually shorter.

— 2020 Scrum Guide page 9

You are guaranteed to get questions about the timebox length of every event on the Scrum Master certification exam. Remember that the Sprint Review is timeboxed to a maximum of 4 hours.

## Test Yourself

Which of the following Scrum events are timeboxed to 4 hours? (Choose 1)

- A) The Sprint Review only
- B) The Sprint Retrospective only
- C) Sprint Planning only
- D) The Sprint Review and the Sprint Retrospective
- E) The Sprint Review and Sprint Planning

Option A is correct. Only the Sprint Review is timeboxed to 4 hours.

The Sprint Retrospective is 3 hours. The Sprint Planning event is capped at 8 hours, and the Daily Scrum is 15 minutes.



Figure 28. Death by PowerPoint is not a great way to run a Sprint Review.

## The Sprint Review is not Simply a Presentation

The Sprint Review is a working session and the Scrum Team should avoid limiting it to a presentation.

— 2020 Scrum Guide page 9

The Scrum Master certification exam will likely include a question or two where one of the incorrect options implies that the purpose of the Sprint Review is to simply do a *demo* or deliver a *presentation*. Don't fall for that bait.

The Sprint Review is much more than just a presentation to stakeholders.

The purpose of the Sprint Review is to inspect the work that was performed during the Sprint and adapt according to feedback received from the stakeholders. At the end of a Sprint Review, the stakeholders should know what was done during the Sprint, and the Scrum Team should have a better idea of what they should be working on next.

When the Sprint Review is complete, it's time for the team to do a Sprint Retrospective, after which the Sprint comes to an end.

Be @Scrumtuous and Follow @Xennial on Twitter

## Chapter 13: Sprint Retrospective

When the work is done, and the Sprint Review is complete, it's time for the Scrum Team to circle the wagons and chat a bit about what went right, what went wrong, and what they want to change when the next Sprint begins.

The purpose of the Sprint Retrospective is to plan ways to increase quality and effectiveness.

The Scrum Team inspects how the last Sprint concerning:

- individuals
- interactions
- processes
- tools
- the Definition of Done.

Inspected elements often vary with the domain of work.

— 2020 Scrum Guide page 10

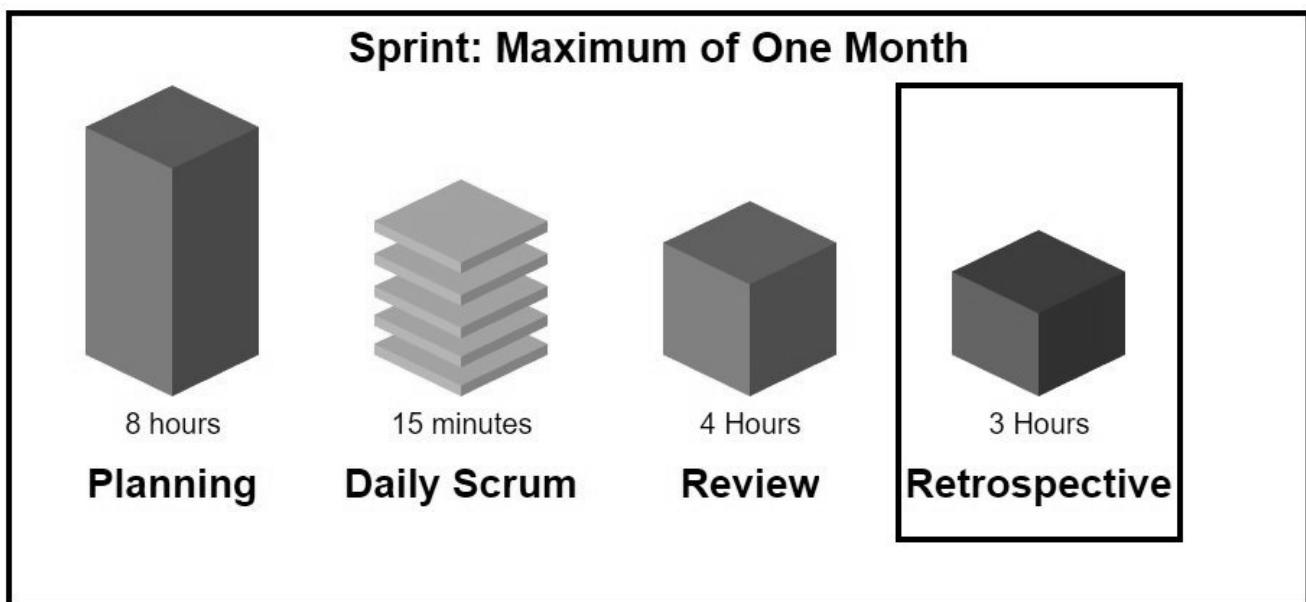


Figure 29. The Sprint Retrospective is the last Scrum event to take place within the Sprint.

## Updating the Sprint Length and the DoD

One question that comes up regularly on the Scrum Master certification exam is *when should the Definition of Done be discussed or updated*. This paragraph makes it clear that the Sprint Retrospective is the correct time to do so.

It's not stated explicitly, but the Sprint Retrospective is also the right time to discuss changing the Sprint length if that's desired. That way if a change is agreed upon, the change can be applied to the next Sprint that starts when the retrospective ends.

The Scrum Guide also indicates that discussions about how tools are working, how people are getting along, and how well processes are implemented within the Scrum framework are best discussed during the Sprint Retrospective.

## Scrum Never Stifles Communication

If the certification exam ever asks when developers should talk about how well their tools are working, or how well they have applied the Scrum framework, the correct answer will be the Sprint Retrospective, not the Daily Scrum.

This is a bit counter-intuitive because you would think developers would be encouraged to talk about their tools and their processes when they're all together in the Daily Scrum, but if it's a meta-discussion about how things are working, that type of discussion belongs in the retrospective.

If a given tool is causing an immediate problem and creating a blocker or impediment that puts the Sprint Goal at risk, then the discussion would be more apt to take place during the Daily Scrum.

## Test Yourself

The developers are not happy with how the Sprint framework has been implemented. When is the best time to discuss this? (Choose 1)

- A) During the Sprint Review
- B) During the Daily Scrum
- C) During Sprint Planning
- D) During the Sprint Retrospective
- E) During their annual review

Option D is correct.

Discussions about how things are working, how various processes are being implemented, and even discussions about how to do Scrum better should happen during the Sprint Retrospective.

## The 80/20 Principle

Assumptions that led them astray are identified and their origins explored.

The Scrum Team discusses:

- what went well during the Sprint,
- what problems it encountered, and
- how those problems were or were not solved.

The Scrum Team identifies the most helpful changes to improve its effectiveness.

The most impactful improvements are addressed as soon as possible.

They may even be added to the Sprint Backlog for the next Sprint.

— 2020 Scrum Guide page 10

This is a weird section. It's the most wishy-washy section of the entire Scrum Guide.

Terms like 'most helpful', 'most impactful', and 'may even be' don't provide much clarity. It's strange because the Scrum Guide is usually very precise about the terms and phrases it uses.

I think the guide is just reflecting the 80/20 principle that asserts a few of your best ideas will end up having the largest impact, so don't feel the need to include every single suggestion made during the Sprint Retrospective in the next Sprint.

The idea is to act upon the best suggestions for improvement right away.

## Adding to the Next Sprint's Backlog?

They may even be added to the Sprint Backlog for the next Sprint.

— 2020 Scrum Guide page 10

That's an interesting line.

In Scrum, we're never really supposed to look beyond the current Sprint.

We don't plan for future Sprints before they start because we never really know how conditions are going to change. One of the Scrum values is the *focus*.

Here is one of the rare instances in the Scrum Guide where something generated in one Sprint is added directly to the next Sprint.

It was stated earlier in this book that when the team goes into the Sprint Planning meeting, the Sprint Backlog is empty, as all unfinished Sprint Backlog items are returned to the Product Backlog when the previous Sprint ends. This is the only instance I can think of where a Sprint Backlog item can be carried over from the current Sprint to the next.

## Test Yourself

Focus is an important part of the Scrum framework. Scrum Teams are expected to stay focused on the current Sprint.

When during the current Sprint are developers encouraged to add items to the next Sprint's Backlog? (Choose 1)

- A) During Sprint planning before the Sprint begins
- B) During the Sprint Review as they collaborate on what to do next
- C) During the Sprint Retrospective as the Scrum Team identifies ways to improve its effectiveness
- D) Scrum Teams should never look beyond the timeline of the current Sprint

Option C is correct.

During the Sprint Retrospective, the Scrum Team may add strategies for improvement to the next Sprint's Backlog.

## Test Yourself

The Scrum Team has come up with two great ideas on how to improve their processes during the Sprint Retrospective. What should they do with these ideas? (Choose 1)

- A) Add them to the current Sprint Backlog
- B) Add them to the current Product Backlog
- C) Add them to the next Product's Product Backlog
- D) Add them to the next Sprint's Sprint Backlog

Option D is correct.

According to the Scrum Guide, great ideas that come out in the Sprint Retrospective can be placed in the Sprint Backlog of the next Sprint.

## The Conclusion of the Sprint

The Sprint Retrospective concludes the Sprint.

It is timeboxed to a maximum of three hours for a one-month Sprint.

For shorter Sprints, the event is usually shorter.

— 2020 Scrum Guide page 10

The Sprint Retrospective is the last timeboxed event mentioned in the Scrum Guide. If you're keeping track, the length of the events in Scrum are:

- **Sprint:** up to 1 month
- **Planning:** up to 8 hours
- **Review:** up to 4 hours
- **Retrospective:** up to 3 hours

While a Sprint is given a fixed time that cannot be extended while a Sprint is in progress, the Scrum Guide never actually refers to a Sprint as a 'timeboxed' event.

And yes, I do realize this is probably the *umpteenth* time I've mentioned the duration of each of the Scrum events in this book. That's because you're likely going to get five or six questions on the topic, so it's not a topic to treat trivially.

## After the Sprint

Once the Sprint Retrospective ends, the Sprint is over and the next Sprint begins.

Every event happens within the scope of a Sprint in Scrum. When one Sprint ends, the next one starts immediately. The end of the Sprint Retrospective is the official end of the Sprint.

## Test Yourself

Which Scrum event marks the end of the Sprint? (Choose 1)

- A) Sprint Planning
- B) Sprint Review
- C) Sprint Retrospective
- D) Sprint Deployment

Option C is correct.

The final event in the Sprint is the Sprint Retrospective.

## Test Yourself

A new Sprint starts: (Choose 1)

- A) Immediately after Sprint Planning
- B) Immediately after the Sprint Review
- C) Immediately after the Sprint Retrospective
- D) Immediately after deployment of the Increment

Option C is correct.

A new Sprint begins immediately after the Sprint Retrospective is over.

## Test Yourself

What is the correct order, from longest to shortest? (Choose 1)

- A) Sprint, Review, Retrospective, Planning
- B) Sprint, Planning, Review, Retrospective
- C) Sprint, Planning, Retrospective, Review
- D) Planning, Sprint, Review, Retrospective
- E) Planning, Review, Retrospective, Sprint

Option B is correct.

The Daily Scrum was left out of these options. It is the shortest of all Scrum Events at 15 minutes.

## Test Yourself

In what order do the Scrum events run? (Choose 1)

- A) Sprint, Review, Retrospective, Planning
- B) Sprint, Planning, Review, Retrospective
- C) Sprint, Planning, Retrospective, Review
- D) Planning, Sprint, Review, Retrospective
- E) Planning, Review, Retrospective, Sprint

Option B is correct.

Everything happens within a Sprint.

Once the Sprint starts, planning occurs. Then work is done and a review is performed. Then a retrospective allows the team to discuss ways to improve for the next Sprint, which starts when the Sprint Retrospective ends.

Be @Scrumtuous and Follow @Xennial on Twitter

## Chapter 14: The Scrum Artifacts

There are three Scrum artifacts:

- The Product Backlog
- The Sprint Backlog
- The Increment of work the developers create

Scrum's artifacts represent work or value.

They are designed to maximize the transparency of key information.

Thus, everyone inspecting them has the same basis for adaptation.

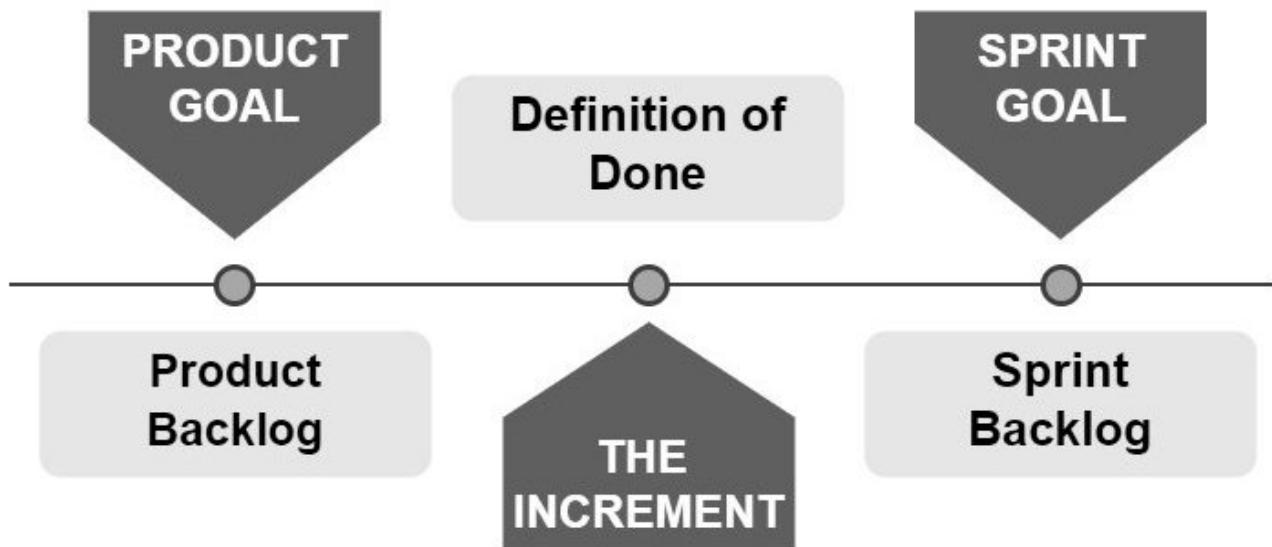
— 2020 Scrum Guide page 10

### Answering Three Important Questions

The existence of these artifacts helps provide transparency by answering the three questions stakeholders constantly want answers to:

- What is currently being done? That's answered by the Sprint Backlog
- What needs to be done? That's answered by the Product Backlog
- What's been done? That's answered by the Increments of work that have been produced

Stakeholders get angry and upset when they don't have answers to these questions. The existence of transparent and inspectable Scrum artifacts tends to calm the nerves of nervous stakeholders.



*Figure 30. Each Scrum artifact, namely the Product Backlog, Sprint Backlog and the Increment, has a corresponding Scrum commitment. The Product Goal, Sprint Goal and the Increment are the Scrum commitments.*

## Test Yourself

Stakeholders are waiting on a new feature and they want to know if it's being actively developed this Sprint. Where would stakeholders go to find this information? (Select 1)

- A) The Product Backlog
- B) The Sprint Backlog
- C) The Definition of Done
- D) The Sprint Goal

Option B is correct.

The Sprint Backlog provides transparency into what is being worked on during the current Sprint.

## Scrum Commitments

The *Scrum commitment* is something new that was added to the 2020 Scrum Guide. Each artifact now maps to a Scrum commitment.

Each artifact contains a commitment to ensure it provides information that enhances transparency and focuses against which progress can be measured:

- For the Product Backlog it is the Product Goal.
- For the Sprint Backlog it is the Sprint Goal.
- For the Increment it is the Definition of Done.

These commitments exist to reinforce empiricism and the Scrum values for the Scrum Team and their stakeholders.

— 2020 Scrum Guide page 10

Every arrow needs a target.

Without a target, an archer doesn't know where to aim. Nor would they know if they ever hit their mark.

In Scrum, artifacts are the arrows. Scrum commitments are the targets.

The Scrum Guide demands that work revolves around the following three artifacts:

- A Product Backlog that represents every feature to be developed;
- A list of backlog items selected for the Sprint; and
- Increments of progress that stakeholders can regularly review

## Why do we need Scrum commitments?

A Scrum commitment is the evaluation criteria against which a Scrum artifact is measured. Scrum commitments improve transparency and enhance a Scrum Team's focus as the project progresses.

Here's how the three Scrum commitments match up to the three Scrum artifacts mentioned above:

- The Product Backlog's Scrum commitment is to the Product Goal.
- The Sprint Backlog's Scrum commitment is to the Sprint Goal.
- The Increment's Scrum commitment is to the Definition of Done.

Agile practitioners don't like plans. In fact, "responding to change over following a plan" is one of the four declared values stated in the Agile Manifesto.

Plans aren't inherently bad. In Scrum, every Sprint begins with a Sprint Planning session. Planning is a requirement of the Scrum framework. Not planning is definitely an anti-pattern.

The problem is that when conscientious, well-intentioned people get a plan, they do their best to follow it. Sometimes that is counter-productive, especially when conditions change, which they inevitably do.



*Figure 31. Scrum commitments are the targets for the Scrum artifacts to aim at.*

## Scrum Commitments, Goals and Plans

Compared to a plan, a Scrum commitment is a higher-order calling to which team members aspire. Goal-driven work is just a better motivator than plan-driven work.

The main purpose of the three commitments is to provide clear, shared, and agreed upon goals that guide all work. The commitments also encourage participants to constantly evaluate how effectively their planned work will help the team follow through on a given commitment.

If the work, plan and commitment are misaligned, developers should sound the alarm and change the plan, because the commitment represents a higher calling than the plan.

## Empiricism and the Scrum Commitments

The Scrum framework's three pillars of empiricism are:

- Transparency
- Inspection
- Adaptation

With a clear Product Goal, well-defined Sprint Goal and a concise Definition of Done, the product development process is more transparent and becomes more Agile.

This allows stakeholders to objectively inspect the project's progress and immediately change and adapt when individual efforts don't align with the team's greater commitments.

Before an archer pulls back her bow, she aims at a target.

For Scrum practitioners, the Scrum commitment is the target to which their artifacts aim.

### Test Yourself

The goal of every Sprint is to create an Increment that lives up to what ultimate measure?  
(Choose 1)

- A) Standard performance metrics
- B) Quality Assurance review
- C) The Definition of Done
- D) A conflict-free merge into the master branch

C is correct.

The Increment commits to measuring up to and conforming to the Definition of Done.

### Test Yourself

How do the Scrum artifacts reinforce empiricism for stakeholders? (Choose 1)

- A) They help the stakeholder's chart velocity
- B) They help Quality Assurance teams confirm quality metrics are achieved
- C) They provide insight into what has been done and what is being done
- D) They help stakeholders estimate the overall cost of the project

Option C is correct.

Empiricism is all about observation and experience. The Scrum artifacts allow stakeholders to know precisely what has been done and exactly what is currently being worked on.

This provides more empirical evidence of the state of the project than burndown charts or cumulative flow diagrams.

---

## Test Yourself

The Scrum artifacts exist to reinforce empiricism only for the Scrum Team.

- True
- False

This is false. The Scrum artifacts are accessible to the stakeholders as well as the Scrum Team. They help reinforce empiricism for everyone involved in the project.



## Chapter 15: The Product Backlog

The Product Owner is the one member of the Scrum Team who is tasked solely with the job of 'optimizing the value of the work done by the Scrum Team.' The way the Product Owner does that is by keeping an ordered list of the most important features and improvements the product under development requires. This list is known as the Product Backlog.

The Product Backlog is an emergent, ordered list of what is needed to improve the product.

— 2020 Scrum Guide page 10

The Scrum Guide's definition makes it clear that the Product Backlog is the list of features that need to be built to improve the product under development. The developers on a Scrum Team should never be working on anything that isn't part of the Product Backlog.



*Figure 32. The Product Owner is solely accountable for the Product Backlog.*

### An Ordered List of Features

The Scrum Guide's definition of the Product Backlog demands that the list be ordered in a sequence that puts the important items at the top.

There is also only one Product Backlog. Any product feature that's needed and likely to be built within a reasonable time horizon is listed in the one, true Product Backlog.

Items that have existed in the backlog for an extended period of time without having been built, or items that are not likely to be built any time soon should be pruned away. The Product Backlog should help provide focus. It shouldn't be cluttered with items that are never likely to be built.

If there's a product enhancement that needs doing, it exists in the Product Backlog.

The Product Backlog will change over time as more is known about what the stakeholders want and what the client needs. That's the idea of the Product Backlog being 'emergent.'

## Single Source of Work

The Product Backlog is the single source of work undertaken by the Scrum Team.

— 2020 Scrum Guide page 10

The fact that the Product Backlog is the *single source* of work to be undertaken by the Scrum Team reinforces the fact that it is also the *only* source of product development work to be undertaken by the Scrum Team.

If the Scrum Team is going to work on something, then it has to first be part of the Product Backlog.

## Test Yourself

Only the Scrum Master can assign work to the Scrum Developers that is not a part of the Product Backlog.

- True
- False

This is false for a number of reasons.

First, the Scrum Master doesn't assign anyone any work, ever. A Scrum Master might coach a Product Owner on how to do their job better, but they never assign anyone work.

Secondly, the Product Backlog is the only source of work for the Scrum Team. If it's not part of the Product Backlog, the Scrum Developers shouldn't be working on it.

## Product Backlog Refinement

Product Backlog items that can be done by the Scrum Team within one Sprint are deemed ready for selection in a Sprint Planning event. They usually acquire this degree of transparency after refining activities.

Product Backlog refinement is the act of breaking down and further defining Product Backlog items into smaller more precise items. This is an ongoing activity to add details, such as a description, order, and size.

Attributes often vary with the domain of work.

— 2020 Scrum Guide page 10

Developers may see a high-priority Product Backlog item that is far too complex to complete in a 3 or 4-week Sprint. When this happens, the developers work with the Product Owner to break the item down into something more manageable.

The goal is always to break a Product Backlog item (PBIs) into something that can be completed within a Sprint. Breaking Product Backlog items down into a day of effort or less can often help with planning and goal setting, although there's no hard and fast rule that says a PBIs should be completable within a workday.

For each selected Product Backlog item, the Developers plan the work necessary to create an Increment that meets the Definition of Done. This is often done by decomposing Product Backlog items into smaller work items of one day or less.

— Sprint Planning page 8

## Test Yourself

Breaking Product Backlog items selected for the Sprint down into smaller work items, sometimes of one day of effort or less, is called: (Choose 1)

- A) Grooming
- B) Refinement
- C) Decomposition
- D) Planning

Option C is correct.

The Scrum Guide describes decomposition as breaking "Product Backlog items into smaller work items of one day or less." The term 'grooming' has fallen out of favor.

## Test Yourself

A Product Backlog item has remained untouched for over a year, and there are no plans for the developers to work on this item within the next 12 months. What should the Product Owner do with this Product Backlog item? (Choose 1)

- A) Remove the Product Backlog item from the Product Backlog
- B) Create a separate, low-priority Product Backlog to track items like this
- C) Assign a high-priority status to the item so it gets completed in an upcoming Sprint
- D) Leave the item in the Product Backlog with a low priority

Option A is correct.

A Product Backlog cluttered with items with no real viable prospect of being worked on within the foreseeable future should be removed from the Product Backlog.

The Product Backlog should help the team focus. It shouldn't create confusion or anxiety.

## Test Yourself

When does Product Backlog refinement occur: (Choose 1)

- A) Product Backlog refinement happens during Sprint Planning
- B) Product Backlog refinement happens during the Sprint Retrospective
- C) Product Backlog refinement happens during the Daily Scrum
- D) Product Backlog refinement is an ongoing activity that happens throughout the Sprint

Option D is correct.

The Scrum Guide has very few rules about what should happen and when. If a Product Backlog item needs more details, needs to be refined, needs to be decomposed, or anything else, then the team should do it right away.

When work needs to be done, don't wait for an official Scrum event to do it. Just get the work done.

## Estimation and Sizing

The Developers who will be doing the work are responsible for the sizing.

The Product Owner may influence the Developers by helping them understand and select trade-offs.

— 2020 Scrum Guide page 10

This point keeps coming up and up again in the Scrum Guide, and you can expect it to come up again and again on the certification exam.

Only the developers know what it takes to accomplish a given piece of work. Only the developer can size up Product Backlog items and estimate how much time a given feature will take to complete.

Do you tell a nuclear engineer how long it should take to cool down a nuclear reactor? No, because you don't know anything about nuclear reactors. They do.

Developers do the estimating in Scrum.

## Test Yourself

Who is responsible for estimating how many Product Backlog items can be completed in a Sprint? (Choose 1)

- A) The Product Owner
- B) The Scrum Master
- C) The Scrum developers
- D) The stakeholders

Option C is correct.

The developers are the experts. Only the developers know how long it will take to complete a Product Backlog item.

## Test Yourself

Taking into account upcoming vacation time is more empirical than estimating productivity based on burndown charts.

- True
- False

This is true.

Burndown charts and velocity calculations are great, but they are not a replacement for actual knowledge and real-world experience.

## Commitment: The Product Goal

The Product Goal describes the future state of the product which can serve as a target for the Scrum Team to plan against.

— 2020 Scrum Guide page 11

Every arrow needs a target.

As the Scrum Team works to build the product, they need to know what the product is that they're building. The Product Goal serves this purpose. It's the Product Backlog's target.

## Test Yourself

The Product Goal represents the current state of the project?

- True
- False

This is false.

The Product Goal represents the future state of the project.



Figure 33. The Product Goal is the target to which the Product Backlog aims at.

## Product Goal and the Product Backlog

The Product Goal is in the Product Backlog.

The rest of the Product Backlog emerges to define "what" will fulfill the Product Goal.

— 2020 Scrum Guide page 11

The Product Goal is a semi-finalized, permanent part of the Product Backlog.

It's semi-finalized because it shouldn't be changing very often. It should represent the vision of the Product Owner and act as a steady target for all members of the Scrum Team to aim at.

On the other hand, the Product Backlog is *emergent*.

The project may start with little more than a vague idea of how to implement the product vision.

The Product Backlog items may be poorly defined, as a full appreciation for exactly what product features are necessary may not exist when the project kicks off.

Over time, as more is learned about the product, the stakeholders, the tech stack, the timelines and the team's capacity, Product Backlog items will be added, refined, decomposed and potentially even deleted.

The Product Goal represents a stable vision of what the Product Owner would like to build. The Product Backlog emerges and is refined over time.

## Value Delivery

A product is a vehicle to deliver value.

A product has:

- a clear boundary,
- known stakeholders,
- well-defined users or customers.

A product could be a service, a physical product, or something more abstract.

— 2020 Product Backlog Definition page 11

The people behind the Scrum Guide have worked hard to get people out of the mindset that Scrum is just for software development.

This section further emphasizes the point that Scrum can be used to develop more than just software.

## Test Yourself

During the Sprint, the product being built will have: (Choose 2)

- A) An approved budget
- B) Known stakeholders
- C) Well-defined users
- D) Potentially unclear boundaries

Options B and C are correct.

The Scrum Guide does not say anything about the topic of budgets, so option A is correct.

Option D is incorrect because the Scrum Guide insists that the product being built must have clear boundaries.

## A Single, Shared Objective

The Product Goal is the long-term objective for the Scrum Team.

They must fulfill (or abandon) one objective before taking on the next.

— 2020 Product Backlog Definition page 11

According to the Scrum Guide, a Scrum Team can only work on one project at a time. They can't have their efforts split between two separate projects.

That doesn't mean an *individual developer* or *Scrum Master* can't be on multiple teams at the same time.

I've quite often seen UI designers who have only a small part to play on a set of active projects be part of multiple Scrum Teams at the same time.

It's also very common for Scrum Masters to be on three or four different teams at a time.

There is no rule against an individual being on multiple teams with separate objectives, but a single Scrum Team can only have one objective in mind.

### Test Yourself!

The Product Owner believes the Scrum Team has enough capacity to work on two separate projects concurrently. How do you, as the Scrum Master, advise the Product Owner to go forward with this plan? (Choose 1)

- Double the timebox for all Scrum events
- Create a second Product Backlog for the new project
- Add features for the new Project into the current Product Backlog
- Inform the Product Owner that a Scrum team can only work on one objective at a time

Option D is correct.

A Scrum Team must fulfill or abandon one objective before taking on the next.

A Scrum Team can have only one objective at a time.

**TIP**

The certification exam will create scenarios that make it sound like it's okay for a Scrum Master or CEO to add items to the Product Backlog. That can never happen. Only the PO can add items to the Product Backlog.

## Test Yourself

The Product Backlog is an ordered list of the things that are needed to improve the product.

- True
- False

This is true.

This precisely describes the purpose of the Product Backlog.

---



## Chapter 16: The Sprint Backlog

If you're ever asked about the Sprint Backlog on the Scrum Master certification exam, there's a good chance the answer is 'The Developers.' The Developers own the Sprint Backlog.

The Sprint Backlog is composed of:

- the set of Product Backlog items selected for the Sprint (what)
- the Sprint Goal (why)
- an actionable plan for delivering the Increment (how)

The Sprint Backlog is a plan by and for the Developers.

— 2020 Scrum Guide page 11

### 3 Questions to Answer

The Sprint Backlog answers three important questions:

- What?
- Why?
- How?

The 'what' is all the items the team hopes to complete during the Sprint. That's all the Product Backlog items the team has 'pulled' and made part of the Sprint Backlog.

The 'why' is the Sprint Goal, which acts as a target for the team to aim at during the Sprint.

The 'how' is the Development team's plan for delivering at least one, and hopefully multiple, usable and valuable Increments of work during the Sprint.

## Who Owns the Sprint Backlog?

The Developers on the Scrum Team completely control the Sprint Backlog.

The Product Owner cannot add to it and the Scrum Master can't delete anything from it.

The Developers completely control the Sprint Backlog. Contrast that against the Product Backlog which is entirely controlled by the Product Owner.

What does that leave the Scrum Master to control? Very little.

Other than a responsibility to guide the team and organization on how to properly apply the Scrum framework, the Scrum Master has very little *control* over the Scrum Team, and virtually no control over the various Scrum artifacts.

On a really well experienced Scrum Team, the Scrum Master has very little work to do. In mature organizations that have been doing Scrum for a long time, a Scrum Master can easily split their efforts across multiple teams. A great Scrum Master eventually coaches themselves out of a job, as the organization becomes so good at Scrum, they hardly need the Scrum Master at all.

## Test Yourself

The Sprint Backlog answers which three questions?

- A) When, where, and why
- B) Who, when, and where
- C) Why, what, and how
- D) How, what, and where

Option C is correct.

The Sprint Backlog answers the questions of why, what, and how. *Who and where* is not a concern of the Sprint Backlog.

## What gets Updated Throughout the Sprint

Scrum is all about constantly inspecting and adapting. As such, the team is allowed to adjust their Sprint plan whenever conditions change, just so long as the changes don't risk the Sprint Goal.

Developers can add or remove items from the Sprint Backlog any time they like.

Quite often teams choose too much work, or even not enough work, and need to either pull things in or put things back into the Product Backlog. Messing around with the Sprint Backlog is 'fair dinkum', so long as the Sprint Goal is not put at risk.

Every team must be given the freedom to change their plans when necessary.

What can't change? The Sprint Goal.

## The Sprint Goal Never Changes After Planning

The Sprint Goal is never allowed to change throughout the Sprint. Once that's finalized in the Sprint Planning meeting, it remains unchanged until the Sprint is over.

The Sprint Backlog is a highly visible, real-time picture of the work that the Developers plan to accomplish during the Sprint to achieve the Sprint Goal.

Consequently, the Sprint Backlog is updated throughout the Sprint as more is learned.

It should have enough detail so that they can inspect their progress in the Daily Scrum.

— 2020 Scrum Guide page 11

## Transparency, Inspection, and Adaptation

One of the three pillars of Scrum is transparency.

The Sprint Backlog helps to create transparency by allowing everyone on the Scrum Team, along with all of the stakeholders, to know exactly what's going on during a Sprint.

Management should never pester the Scrum Team with requests for status meetings or regular updates through email. The Sprint Backlog should transparently answer any questions stakeholders have about what is being developed and how the development process is moving along.

### Test Yourself

When should the Sprint Backlog be updated? (Choose 1)

- A) During Sprint planning when more details emerge
- B) During the Daily Scrum as the Scrum developers adapt
- C) During the Sprint Review as stakeholders give feedback
- D) During the Sprint any time more is learned

Option D is correct.

The Sprint Backlog should be an up-to-date reflection of what is being done and what has been done.

It should be updated regularly.

## Sprint Goal

Every arrow needs a target to aim at. For the Sprint, the target is the Sprint Goal.

The Sprint Goal is the single objective for the Sprint. Although the Sprint Goal is a commitment by the Developers, it provides flexibility in terms of the exact work needed to achieve it. The Sprint Goal also creates coherence and focus, encouraging the Scrum Team to work together rather than on separate initiatives.

— 2020 Scrum Guide page 11

The Sprint Goal is the commitment the Sprint Backlog makes.

The Sprint Goal describes what the developers are trying to achieve in the current Sprint. It gives the team something to aim at.

Now let's make one thing clear - the development team doesn't always achieve the Sprint Goal, and that's okay.

Some teams might not achieve the Sprint Goal the majority of the time. If that's a problem, then maybe it's something the team can talk about in the Sprint Retrospective. But at the same time, it might not be a problem.

What matters is that the Sprint Goal drives the team forward. So long as a usable increment of work is produced at the end of the Sprint and progress is made towards building the product, a missed goal isn't the end of the world.

## Test Yourself

Issues beyond the Scrum Team's control have made it almost impossible to achieve the Sprint Goal, and there's still a week left in the Sprint. What should the Scrum Master do? (Choose 1)

- A) Change the Sprint Goal
- B) Cancel the Sprint
- C) Schedule a mid-Sprint planning session
- D) Have the team continue to work towards the goal

The Sprint can't be canceled unless the Sprint Goal becomes obsolete, and that's a rare occurrence.

You're also not allowed to change the Sprint Goal once it's finalized.

The correct answer here is to just keep working towards the Sprint Goal.

It's not the end of the world if the Sprint Goal is not achieved. Sprints are short. You can create a new Sprint Goal when the current Sprint is finished.

## Product Owner and Developer Collaboration

If the Developers have bitten off more than they can chew, and they don't think they can complete all of the tasks they've assigned to themselves, they can remove items from the Sprint Backlog.

When this happens, developers should inform the Product Owner. If selected items aren't going to get finished during the Sprint, the Product Owner should know.

Furthermore, the Product Owner knows which Product Backlog items are most important, so the PO might have a better insights into which backlog items should remain in the Sprint and which ones shouldn't.

The Sprint Goal is created during the Sprint Planning event and then added to the Sprint Backlog. As the Developers work during the Sprint, they keep the Sprint Goal in mind.

If the work turns out to be different than they expected, they collaborate with the Product Owner to negotiate the scope of the Sprint Backlog within the Sprint without affecting the Sprint Goal.

— 2020 Scrum Guide page 11

It's not unusual for teams to overestimate how much work they can accomplish during a Sprint.

If developers need to decompose backlog items or even remove items from the Sprint Backlog, that's fine, so long as all of this happens without risking the Sprint Goal.

### Test Yourself

If Developers find the scope of work scheduled for a Sprint is too much, with whom should they negotiate the Sprint Backlog's scope? (Choose 1)

- A) The Product Owner
- B) The Scrum Master
- C) Their fellow developers
- D) The stakeholders.

Option A is correct.

Any time the developers need to scale back on their work or break down Product Backlog items into smaller pieces, it's always wise to speak with the Product Owner for clarification. The Product Owner can also help developers understand which product features should be included in order to not to put the Sprint Goal at risk.

**TIP**

Developers wholly control the Sprint Backlog. Only the Scrum Developers can add to it or delete from it. However, the Sprint Goal can't change during the Sprint.



## Chapter 17: The Increment

Scrum is an iterative and incremental framework.

We see how Scrum is iterative through the cycling of Sprint after Sprint after Sprint. Within those Sprints, the same events occur over and over again.

But how is Scrum incremental?

### Iterative and Incremental

In Scrum, we add to the product every Sprint. Every Sprint is supposed to deliver an Increment that is *useful* and *valuable*. You never do a Sprint that doesn't try to create something useful.

There are no 'infrastructure Sprints' in Scrum where you just make sure everyone has the right software installed and access to the JIRA board.

In Scrum, every Sprint has to create something of value that adds to the existing product in a useful and meaningful way.

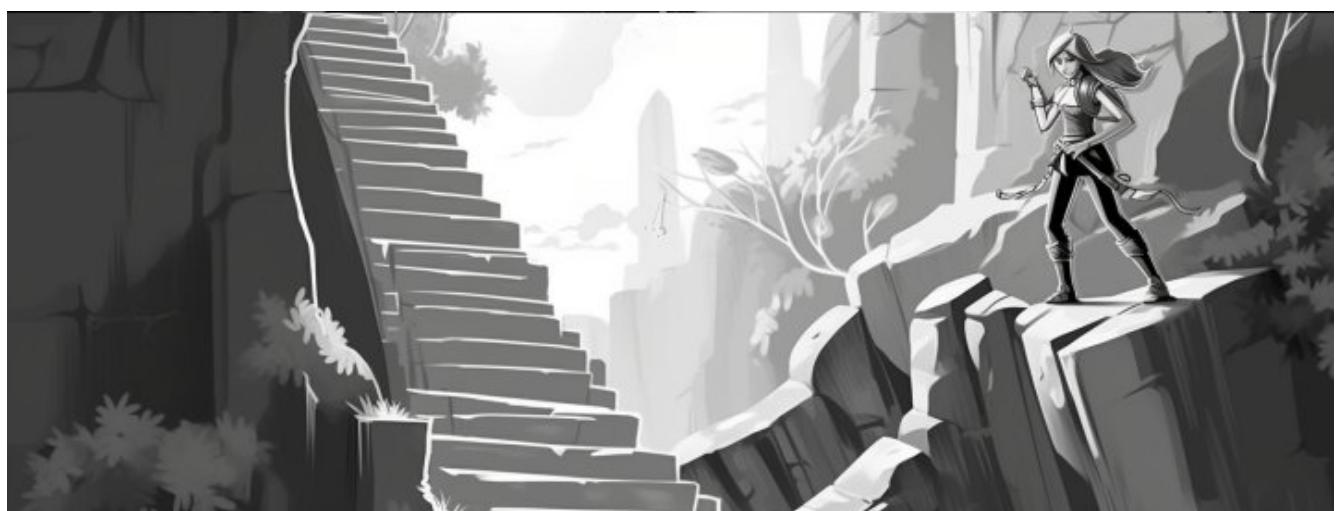
That's how Scrum is incremental. Every Sprint adds to what was created in the Sprints prior. That's the objective of every Sprint.

An Increment is a concrete stepping stone toward the Product Goal.

Each Increment is additive to all prior Increments and thoroughly verified, ensuring that all Increments work together.

To provide value, the Increment must be usable.

— 2020 Scrum Guide page 11



*Figure 34. Like steps in a staircase, each Increment builds on top of previous increments. We move forward by iteratively creating valuable and usable increments that take us progressively closer to the Product Goal.*

## Integrated Increments

An Increment must be added to what's already been done, and it must be able to fully integrate and work with all previous increments. If an Increment isn't compatible with past Increments, then it's not a valid Increment. If an Increment stands alone and doesn't increase the value of the product built to date, then it's not valid.

On the certification exam, there will be many questions about *integration*, which is intimidating, because there is no talk in the Scrum Guide about integration.

- There are no integration teams in Scrum
- There are no specific Scrum events where integration happens
- There is no designated time for integration to happen

The words 'integrate' and 'integration' never appear once in the 2020 Scrum Guide.

## Understanding Integration

However, integration is important when it comes to the creation of an Increment. Whatever you create today must 'integrate with' what you created yesterday for it to be a valid Increment.

Many questions on the Scrum Master certification exam will reference *integration*.

Software developers get tripped up on the word 'integration' because they get visions of merging code or fixing Git conflicts or doing GitHub pull requests. Don't confuse yourself with those things. The idea of 'integration' is much simpler than that.

Just imagine you're a carpenter and you built a door for the house your team is building. The door is currently tied to the roof of your car. Is it integrated with the house?

No, a door sitting on the roof of your car that is supposed to go on the house you're building is not integrated with the house if it's strapped to the roof of your car.

When the door is put on its hinges, attached to the house, and it can swing open and closed freely then it can be considered *integrated*. When integrated, it becomes a useful, valuable Increment.

By putting the door on the house, you have integrated your work into the product and you have added a valuable Increment of work that stakeholders can see, touch, feel, and even criticize.

## A Poorly Integrated Increment

Now imagine you took the door off the roof of your car, tried to attach it to the house and:

- The door is too large
- The hinges are incompatible
- You realize the door is the wrong color

You've done work on the door, there's no arguing about that. But has that work been integrated?

Did you just create a usable Increment of work that adds to the existing project? No, you didn't, because your door is useless.



*Figure 35. If a house needs a door, and you have the door at the ready, you don't wait a week to integrate it.*

## When to Integrate

Let's go back to the super, amazing door you created. When should you attach your perfect door to the house?

The Scrum Master exam will often ask you when you should integrate your work into the existing project. People get all flustered with these questions, wondering when the best time is to integrate your work.

Don't get flustered. Get pragmatic.

When you think about integration pragmatically, the questions answer themselves.

## The Right Time to Integrate

So, when should you integrate your work into the existing product?

- During the Daily Scrum?
- During the Sprint Review?
- During the Sprint Retrospective?
- During the Sprint Planning?
- Right now?

'Right now' is the correct answer.

If you've got the perfect door ready to be put on your new house's hinges, you don't sit for two weeks waiting for the Sprint Review to do it. You integrate it right away.

It would be nuts to imagine waiting two weeks to hang the front door on the house, or as we'd say in Scrum, 'integrate your work into the product.'

As soon as your work is done you integrate your work. Then you move on to the next piece of work that needs to be done. You keep doing that over and over and over again until it's time for the Sprint Review.

Scrum prides itself on being based on *lean principles*. Waiting around is not lean. Doing something right away is lean. On the Scrum Master exam, a lean approach to product building is the correct approach.

## Test Yourself

Which one of the following is correct?

- A) Scrum is incremental but not iterative.
- B) Scrum is iterative but not incremental.
- C) Scrum is both incremental and iterative.
- D) Scrum is neither incremental nor iterative.

Option C is correct.

Scrum is iterative in that a set of steps get repeated each Sprint, and it is incremental in that each Sprint produces Increments of work that get added to all past increments.

## Multiple Increments

The goal of each Sprint is to get at least one valuable and usable Increment added to the project, but that shouldn't stop you from integrating as many Increments as you can.

If a carpenter adds a front door to a house, she doesn't wait until the next Sprint to start working on the back door. She gets working on the back door and integrates that into the house as soon she's done with that. Increments should keep being added to the product constantly throughout Sprint.

Multiple Increments may be created within a Sprint.

The sum of the Increments is presented at the Sprint Review thus supporting empiricism.

However, an Increment may be delivered to stakeholders before the end of the Sprint. The Sprint Review should never be considered a gate to releasing value.

Work cannot be considered part of an Increment unless it meets the Definition of Done.

— 2020 Scrum Guide page 12

## Continuous Delivery of Valuable Increments

Some people get the impression that the only time stakeholders should be allowed to see what developers have done is at the Sprint Review.

If developers create an Increment they want the stakeholders to see, nothing is stopping them from presenting it to them at any time throughout the Sprint. And why should a developer wait to show off their amazing work?

If the developers have created something they are proud of, and the stakeholders are anxious to see it, why would the development team wait until the Sprint Review to show it to them?

The Sprint Review makes sure the stakeholders get to see progress at least once throughout a Sprint, but that just ensures a minimum. Developers are always encouraged to be transparent and share their progress with interested stakeholders.

## Commitment: The Definition of Done

The Definition of Done is a formal description of the state of the Increment when it meets the quality measures required for the product.

The moment a Product Backlog item meets the Definition of Done, an Increment is born.

— 2020 Scrum Guide page 12

The Definition of Done must be clear and understood by everyone on the Scrum Team. It must also be understood by the stakeholders and the organization. It creates a common understanding of what has to be done for a feature to be complete.

## Test Yourself

The Definition of Done is an informal understanding of what needs to be done to turn a Product Backlog item into an Increment.

- True
- False

This is false

The Definition of Done is a *formal description*, not an informal understanding.

## Test Yourself

The application has failed to scale beyond a single processing core, and this has caused various production issues.

What is the best way for the Scrum Team to proceed with this issue? (Choose 1)

- A) Assign multi-core processing support to the DevOps team
- B) Create a testing team to implement multi-processor support
- C) Add multi-processor support to the Definition of Done
- D) Inform that the QA team that they should test the app on multiple processors

Option C is correct.

If there is a quality metric that the product must support, it is the development team's responsibility to support it.

If the multi-core requirement is added to the Definition of Done, the developers will not be allowed to integrate or release any increments that don't meet the multi-core requirement.

Remember that there are no sub-teams in Scrum, and there is no talk anywhere in the Scrum Guide about:

- DevOps
- UAT
- QA teams
- Testers

Any time you see these terms on the exam, they are usually indicative of an incorrect answer.

## Scrum Pillars, Transparency, and 'Done'

The Definition of Done creates transparency by providing everyone with a shared understanding of what work was completed as part of the Increment.

If a Product Backlog item does not meet the Definition of Done, it cannot be released or even presented at the Sprint Review. Instead, it returns to the Product Backlog for future consideration.

— 2020 Scrum Guide page 12

Test takers are often killed on this point, so pay attention. When a Product Backlog item is not completed during a Sprint, it is put back into the Product Backlog.

- It's not presented at the Sprint Review
- It is not 'partially presented' at the Sprint Review
- Partial points are not assigned to it (There are no 'points' in Scrum!)
- The feature is not automatically added to the next Sprint's Sprint Backlog
- The feature is not partially released

If a Product Backlog item selected for the Sprint does not meet the Definition of Done by the end of the Sprint, it is thrown back into the Product Backlog as though nobody had ever even been working on it. Even if it's 90% complete, it's thrown back into the Product Backlog for the Product Owner to reprioritize.

### Test Yourself

An important, critical feature is 99% done but it won't be ready for the Sprint Review tomorrow.

What should the team do? (Choose 1)

- A) Present the work that was completed for stakeholders to see at the Sprint Review
- B) Assign 99% of the points to the current Sprint and assign 1% of the points to the next Sprint
- C) Automatically add the Product Backlog item to the next Sprint's Sprint Backlog
- D) Extend the Sprint to allow the developer to get the feature to 100%
- E) Return the item to the Product Backlog and do not present it at the Sprint Review

Option E is correct.

The Scrum Guide is clear. If a feature does not meet the Definition of Done, it is not presented at the Sprint Review, and it is returned to the Product Backlog.

## Organizational Standards

If the Definition of Done for an Increment is part of the standards of the organization, all Scrum Teams must follow it as a minimum.

If it is not an organizational standard, the Scrum Team must create a Definition of Done appropriately for the product.

— 2020 Scrum Guide page 12

Who creates the Definition of Done?

It's created by the Scrum Team unless there is already an existing organizational standard.

If the Scrum Team wants to create a more rigorous Definition of Done, they are certainly allowed to do that, but it must build on top of the one the company already has. It cannot be more lax.

## Done Over Time

One of the interesting things that happens with Scrum Teams as they become more familiar with the framework and the product they are building is they extend and enhance their Definition of Done.

Over time, the Definition will likely become more rigorous, with more quality gates to pass than when the team first adopted Scrum.

Who creates the Definition of Done? (Choose 1)

- A) The Product Owner if there is no organizational standard
- B) The Scrum Master if there is no organizational standard
- C) The Scrum Team if there is no organizational standard
- D) The stakeholders if there is no organizational standard

Option C is correct.

The Scrum Team creates a Definition of Done if an organizational standard does not exist.

## Bad Definitions of Done

Sometimes Scrum Teams will find themselves in a situation where they are dealing with a Definition of Done that is impossible to achieve.

For example, if the Definition of Done says every piece of code has to be tested on a quantum computer with 100,000 qubits, despite the fact that no computer in the world has more than 1000

quibits, then the Definition of Done is too rigorous.

The Definition of Done should not be changed too often, and it's not a good idea to make a Definition of Done less rigorous over time, but if the Definition of Done includes a requirement that is impossible to achieve due to missing hardware, software or infrastructure, then it is permissible to change the Definition of Done to make it possible to deliver a usable Increment.

If the Definition of Done does need to change, such a discussion is best to take place during the Sprint Retrospective.

## Multiple Teams and the Definition of Done

The Developers are required to conform to the Definition of Done.

If multiple Scrum Teams are working together on a product, they must mutually define and comply with the same Definition of Done.

— 2020 Scrum Guide page 12

There are only a few instances in the Scrum Guide where it discusses multiple teams working on the same project.

The rules the Scrum Guide requires for multiple teams working on the same project are few, but they are important and you will be tested on them.

The Scrum Guide demands that multiple teams that work on the same project must:

- Each share the same Product Owner
- Each share the same Product Backlog
- Each share the same Product Goal
- Each share the same Definition of Done

## What Scrum Doesn't Say About Multiple Teams

There is nothing in the Scrum Guide that asserts multiple teams working on the same project must:

- Start their Sprints at the same time
- Have Sprints that are the same length
- Have the same number of team members
- Have the same Scrum Master

There will be several questions about multiple teams working on the same project on the certification exam, and those questions will provide options that seem reasonable but are incorrect.

When answering questions on the certification exam, focus on answers that map as closely as possible to what is explicitly stated in the Scrum Guide.

## Test Yourself

What must be shared between multiple teams working on the same product at the same time? (Choose 2)

- A) The Product Owner
- B) The Scrum Master
- C) The Sprint Backlog
- D) The Product Backlog
- E) Multiple Scrum Teams cannot work on the same product at the same time

Options A and D are correct.

Each Scrum Team has its own Sprint Backlog. A Sprint Backlog is not shared across multiple teams working on the same product.

A Scrum Master can be shared between multiple teams, or a Scrum master can dedicate 100% of their time to one team. The Scrum Guide doesn't advise for or against either scenario.

When multiple teams work on the same project together:

- Each team shares the same Product Owner
- Each team shares the same Product Backlog
- Each team shares the same Product Goal
- Each team shares the same Definition of Done

## PART II: THE EXAM QUESTIONS

The second part of this book is all about testing your knowledge of the Scrum Guide and possibly introducing you to some of the unexpected ways the exam tests your knowledge.

There are five practice exams in this section.

- The last exam is 40 questions, and I believe it is slightly harder than the actual exam.
- The two exams prior to that are 25 questions long and are a good approximation to the Scrum Master certification exam's difficulty
- The two exams that kick off the second half of the book? Well, they're unusual.

Let me explain.



*Figure 36. Do this set of exam questions first. Then, we'll show you a cool trick to help you answer the toughest exam questions.*

### The Secret Test-Taking Trick

There's a trick to answering the toughest questions on the Scrum Master exam, but I'd like you to take the first practice exam first before the trick is revealed.

The questions in the first practice exam fall on the harder side of what to expect on the Scrum Master certification exam, so don't get discouraged if they seem difficult. That's intentional.

When you complete the first practice exam, I'll reveal the trick and test you on a set of questions similar to those in the first exam, but with the trick applied.

I think you'll find it makes a big difference.

Now get started with the practice exams. Good luck!



## Ten Tough Scrum Master Practice Exam Questions

On the actual Scrum Master certification exam, some questions may have multiple correct answers. You will be told how many options to select.

On this practice test, each question has only one answer.

---

### Question 1

Scrum is a proven and complete software development process.

- True
- False

### Question 2

Which of the following statements are true about the Scrum Master role?

- A. The Scrum Master cannot be one of the developers on the team
- B. The Scrum Master cannot also be the team's Product Owner
- C. The Scrum Master cannot be one of the product's stakeholders
- D. The Scrum Master can be a stakeholder and a developer at the same time

### Question 3

The Scrum Team has just been assembled, but the IT department has been slow to provision resources like laptops, software, build tools and test environments.

The team does not want to start the first Sprint next week given the lack of preparedness.

How would you as the Scrum Master recommend the team proceed?

- A. Explain to the organization that 'Sprint 0' exists to allow the team to get prepared for development
- B. Ask the team to work toward the development of a small, valuable increment despite the lack of preparedness
- C. Declare this an infrastructure Sprint where developers simply configure their tools and software
- D. Delay the start of the Sprint until the organization can properly provision the required tools and infrastructure

## Question 4

A developer has completed a feature that when integrated into the codebase will create a valuable and usable increment. Completion of the Sprint Goal depends upon it.

When should the developer integrate their work into the product?

- A. During the Daily Scrum, after achieving consensus with the other developers
- B. After the development Sprint finishes with the consent of the Scrum Master
- C. Immediately so long as the work meets the team's Definition of Done
- D. During the Sprint Review so stakeholders can witness the integration

## Question 5

A developer has identified a troublesome and worrying security vulnerability in the codebase and has asked you, the Scrum Master, what should be done about it.

What guidance should you provide the developer on this topic?

- A. Have the developer make this issue a top priority item in tomorrow's Daily Scrum
- B. Have the developer discuss the issue with the Product Owner and plan to fix the issue in the next Sprint
- C. Have the developer discuss the issue with developers right away
- D. Have the developer raise the issue as a priority item during the next Sprint Planning meeting

## Question 6

Under which circumstance are developers allowed to discuss product development directly with stakeholders?

- A. Developers can only speak directly with stakeholders during the Sprint Retrospective
- B. Developers can only speak directly with stakeholders if the Product Owner is present
- C. Developers can only speak directly with stakeholders with permission from the Scrum Master
- D. Developers are not allowed speak directly with stakeholders
- E. Developers can speak directly with stakeholders at any time

## Question 7

Human Resources has just hired the three new programmers that are required in order to keep development progressing at a sustainable pace. However, these programmers have arrived right in the middle of a four week Sprint.

As the Scrum Master, when do you advise these programmers be added to the development effort?

- A. At the moment they arrive
- B. At the end of the current Sprint but before the next Sprint begins
- C. At the following day's Daily Scrum
- D. During the next Sprint Planning event

## Question 8

Given a well-functioning team of 4 developers, what impact will the addition of 3 new developers likely have on the productivity of the existing team members?

- A. Individual productivity will increase permanently
- B. Individual productivity will increase over the short term
- C. Individual productivity will decrease permanently
- D. Individual productivity will decrease over the short term

## Question 9

Developers have become frustrated with the fact that peripheral tasks like quality control (QA), user acceptance testing (UAT) and documentation have taken away from their core development efforts, subsequently putting the project at risk.

How would you advise the team to move forward?

- A. Have the Scrum team create a sub-team dedicated to QA, UAT testing and documentation
- B. Have the Scrum team outsource QA, UAT and documentation to a third party
- C. Have the Product Owner remove quality control, testing and documentation from the Definition of Done
- D. Advise the Scrum Team they are responsible for all aspects of delivering a product Increment

## Question 10

Unforeseen circumstances have reduced developer productivity and half-way through the Sprint it is clear that the Sprint Plan must change drastically in order for the Sprint Goal to be achieved.

What guidance would you, as the Scrum Master, provide to the team?

- A. Tell the developers to cancel the development Sprint and start a new Sprint Planning session
- B. Have the developers regroup and update the Sprint Plan as soon as possible
- C. Have the developers dedicate time during the next Daily Scrum to update the Sprint Goal
- D. Tell the developers the Sprint Plan cannot change and have them continue working as planned towards the Sprint Goal

# Answers to the First Practice Exam

## Question 1

Scrum is a proven and complete software development process.

- True
- False

This is false for three reasons:

1. Scrum is not a process or a methodology. Scrum is a lightweight framework.
2. Scrum can be used to develop any product, not just a software product.
3. Scrum self-describes as an *incomplete* framework.

"Scrum is a lightweight framework that helps people, teams and organizations generate value through adaptive solutions for complex problems. The Scrum framework is purposefully incomplete."

- The Scrum Guide, page 3.

## Question 2

Which of the following statements are true about the Scrum Master role?

- A. The Scrum Master cannot be one of the developers on the team
- B. The Scrum Master cannot also be the team's Product Owner
- C. The Scrum Master cannot be one of the product's stakeholders
- D. The Scrum Master can be a stakeholder and a developer at the same time

Option D is correct.

There are no rules in the Scrum Guide that say a person cannot take on multiple roles or *accountabilities* at the same time.

In fact, on small startups, it's not uncommon for the Scrum Master to also be a stakeholder while also helping out with development. Furthermore, if the Product Owner got sick, that person might even take on Product Owner accountabilities temporarily as well.

The only rule the Scrum Guide provides on a Scrum Master or Product Owner doing development is that when they do help to build the product, they must participate in the Daily Scrum as a developer.

"If the Product Owner or Scrum Master are actively working on items in the Sprint Backlog, they participate [in the Daily Scrum] as Developers." -The Scrum Guide, page 9.

### Question 3

The Scrum Team has just been assembled, but the IT department has been slow to provision resources like laptops, software, build tools and test environments.

The team does not want to start the first Sprint next week given the lack of preparedness.

How would you as the Scrum Master recommend the team proceed?

- A. Explain to the organization that 'Sprint 0' exists to allow the team to get prepared for development
- B. Ask the team to work toward the development of a small, valuable increment despite the lack of preparedness
- C. Declare this an infrastructure Sprint where developers simply configure their tools and software
- D. Delay the start of the Sprint until the organization can properly provision the required tools and infrastructure

Option B is correct.

There is no such thing as an infrastructure Sprint or a Sprint 0 in Scrum.

Every Sprint must attempt to develop an Increment of value that will be part of the final product.

No organization is ever sufficiently prepared for development to begin.

If Scrum teams waited for all of the tools they require to be provisioned and available before they started to work, it's unlikely that any work would ever get started.

### Question 4

A developer has completed a feature that, when integrated into the codebase, will create a valuable and usable increment. Completion of the Sprint Goal depends upon it.

When should the developer integrate their work into the product?

- A. During the Daily Scrum, after achieving consensus with the other developers
- B. After the development Sprint finishes with the consent of the Scrum Master
- C. Immediately so long as the work meets the team's Definition of Done
- D. During the Sprint Review so stakeholders can witness the integration

Option C is correct.

If a developer completes any work that meets the Definition of Done, it should be immediately integrated into the product.

If work cannot be integrated into the product being built, it is not useful and it does not create a valid Increment. Integration, and ensuring that work does indeed integrate, is essential.

A developer doesn't need anyone's permission to integrate their work into the product.

If there is a piece of work that is complete and can be successfully integrated into the final product, it should be integrated immediately. Waiting serves no purpose, nor is it in line with Scrum's adherence to lean thinking.

## Question 5

A developer has identified a troublesome and worrying security vulnerability in the codebase and has asked you, the Scrum Master, what should be done about it.

What guidance should you provide the developer on this topic?

- A. Have the developer make this issue a top priority item in tomorrow's Daily Scrum
- B. Have the developer discuss the issue with the Product Owner and plan to fix the issue in the next Sprint
- C. Have the developer discuss the issue with developers right away
- D. Have the developer raise the issue as a priority item during the next Sprint Planning meeting

Option C is correct.

If there is a serious security flaw in the codebase, it's an issue that should be raised immediately with the development team.

Scrum provides a number of events, such as the Daily Scrum and the Sprint Retrospective, to encourage discussion and act as feedback loops. However, the existence of these events should never stand in the way of pragmatic or essential discussions between team members taking place.

If an emergency arises, it should be addressed immediately. Don't wait for a scheduled Scrum event to adapt to changing conditions.

## Question 6

Under which circumstance are developers allowed to discuss product development directly with stakeholders?

- A. Developers can only speak directly with stakeholders during the Sprint Retrospective
- B. Developers can only speak directly with stakeholders if the Product Owner is present
- C. Developers can only speak directly with stakeholders with permission from the Scrum Master
- D. Developers are not allowed speak directly with stakeholders
- E. Developers can speak directly with stakeholders at any time

Option E is correct.

There are no rules in Scrum that restricts a developer from talking to a stakeholder.

One of the Scrum pillars is transparency. One of the Scrum values is openness.

What does it say about the transparency and openness of an organization that puts rules and restrictions around when a developer might be allowed to talk to a stakeholder?

The Product Owner is responsible for speaking with stakeholders and discovering which features and functionality they value most.

However, in terms of speaking with developers about the Sprint Backlog, Product Backlog, Sprint Goal or anything else, there is nothing in the Scrum Guide that restricts such discussions. In fact, Scrum encourages it.

## Question 7

Human Resources has just hired the three new programmers that are required in order to keep development progressing at a sustainable pace. However, these programmers have arrived right in the middle of a four week Sprint.

As the Scrum Master, when do you advise these programmers be added to the development effort?

- A. At the moment they arrive
- B. At the end of the current Sprint but before the next Sprint begins
- C. At the following day's Daily Scrum
- D. During the next Sprint Planning event

Option A is correct.

If your project needs programmers to assist in the development effort, why would you wait even one minute to add them to the team and get them contributing to the development effort?

With regards to option B, remember that there is no delay in Scrum between the time when one Sprint ends and the next Sprint begins. When one Sprint ends, the next Sprint starts immediately.

## Question 8

Given a well-functioning team of 4 developers, what impact will the addition of 3 new developers likely have on the productivity of the existing team members?

- A. Individual productivity will increase permanently
- B. Individual productivity will increase over the short term
- C. Individual productivity will decrease permanently
- D. Individual productivity will decrease over the short term

Option D is correct.

The onboarding process of new team members often pulls existing developers away from their normal routines, as they spend time with new devs explaining the product, helping them connect to GitHub, set up a printer and even find the restroom.

Over the long run, individual productivity should go back to normal, but in the short term it will decrease.

## Question 9

Developers have become frustrated with the fact that peripheral tasks like quality control, user acceptance testing and documentation have taken away from their core development efforts, subsequently putting the project at risk.

How would you advise the team to move forward?

- A. Have the Scrum team create a sub-team dedicated to QA, UAT testing and documentation
- B. Have the Scrum team outsource QA, UAT and documentation to a third party
- C. Have the Product Owner remove quality control, testing and documentation from the Definition of Done
- D. Advise the Scrum Team they are responsible for all aspects of delivering a product Increment

Option D is correct.

There are no sub-teams in Scrum, nor are there any hierarchies of developers.

"Within a Scrum Team, there are no sub-teams or hierarchies. It is a cohesive unit of professionals."

- The Scrum Guide, page 5.

If there is a backlog item that requires documentation to be generated, or the Definition of Done requires a set of user acceptance tests (UAT) to be performed, then it is the job of the individuals on the development team to get those tasks done.

The developers on the Scrum Team are responsible for the end-to-end development of all features in the Product Backlog with full compliance to any non-functional requirements the Definition of Done may demand.

## Question 10

Unforeseen circumstances have reduced developer productivity, and half-way through the Sprint it is clear that the Sprint Plan must change drastically in order for the Sprint Goal to be achieved.

What guidance would you, as the Scrum Master, provide to the team?

- A. Tell the developers to cancel the development Sprint and start a new Sprint Planning session
- B. Have the developers regroup and update the Sprint Plan as soon as possible
- C. Have the developers dedicate time during the next Daily Scrum to update the Sprint Goal
- D. Tell the developers the Sprint Plan cannot change and have them continue working as planned towards the Sprint Goal

Option B is correct.

If circumstances change and the developers need to update their plan, they should update their plan immediately.

During the Sprint, the Sprint Goal cannot change, nor can a Sprint be cancelled on the whim of the Scrum Master or the developers. Only a Product Owner can cancel a Sprint, and only if the Sprint Goal has become obsolete.

What *can* change during the Sprint is the Sprint Plan. It's actually expected to be updated throughout the Sprint as conditions change and more is learned about the product being developed.

Developers should certainly not wait for a scheduled event like the Daily Scrum to adapt their plan if it's necessary.

"The Daily Scrum is not the only time Developers are allowed to adjust their plan. They often meet throughout the day for more detailed discussions about adapting or replanning."

-The Scrum Guide, page 9.



## A Simple Trick

Here's a simple trick to help you correctly answer some of the hardest questions you'll face on the Scrum Master certification exam.

I've alluded to this trick throughout the book, but I really want to drive the idea home with the next set of practice exam questions.

## A Hack to Help You Pass the Exam

Most people who take the Scrum Master exam come from the world of software development. That can really clutter your mind when answering a certification exam question.

Instead of thinking about how a given exam question applies to the field of software development, reframe the question in a completely different domain.



*Figure 37. Reframing exam questions outside of software development makes them easier to answer.*

The following exam questions are a mirror image of the first set of questions you were just asked, but reframed in a much more ridiculous and extreme manner.

The scenario used for the next set of questions is that of a shipwreck where survivors from a marooned party boat are forced to build a survival camp as they await rescue.

- The extreme, life-and-death predicament helps to simplify the interpretation of Scrum
- The ridiculousness makes the questions a bit more interesting.

Be @Scrumtuous and Follow @Xennial on Twitter

I think you'll find the next set of questions to be a bit more fun to answer than any of the others previously asked in this book.

You'll also find they give you a deeper understanding of the Scrum framework as well.

## The Party Boat Disaster

A party boat from a popular tech conference was blown off course by a surprise typhoon, and 100 injured developers are now shipwrecked on a deserted island somewhere in the South Pacific.

The ship is marooned on rocky coral, stranded about 30 meters from the island's shore, and most of the survivors are too injured to make it to dry land.

Nobody knows what to do.

Fortunately, one developer took a Scrum course and has a copy of the 2020 Scrum Guide with them.

The castaways are going to make it through this ordeal, and a Scrum based approach to survival will guide them through it.

Their first task? Get a team of developers to go to shore and build a survival camp.



*Figure 38. The following set of questions frame Scrum in a life-or-death survival scenario where castaways from a shipwreck must build a survival shelter.*

## Questions

On the actual Scrum Master certification exam, some questions may have multiple correct answers. You will be told how many options to select.

On this practice test, each question has only one answer.

Good luck!

## Question 1

Scrum is software development process and should not be applied to life-or-death tasks like building a survival shelter.

- True
- False

## Question 2

You, the Scrum Master, have outdoors experience. You would like to be a developer on the first Sprint and help build the temporary shelter.

How should you proceed in order not to violate the rules of the Scrum Guide?

- A. You must temporarily assign Scrum Master duties to another person while you act as a developer
- B. You must temporarily assign Scrum Master duties to the Product Owner while you act as a developer
- C. You must participate in the Daily Scrum during the Sprint
- D. Scrum Masters cannot be a developer on their Scrum Team

## Question 3

A team of four developers, along with a Product Owner and the Scrum Master, have made it to shore and need to start building a survival shelter. However, the team is at odds over what they should do for the very first Sprint.

Whose approach is most in line with the ethos of the Scrum Guide?

- A. Jim wants to declare a 'Sprint 0' where time is spent planning and designing the survival camp
- B. Julie wants to spend the first Sprint searching the island for tools they can use to build the survival camp
- C. Jeff wants to spend the first Sprint searching the island for edible food and clean water
- D. Jessica wants to build a simple communication system from the camp to the ship using mirrors she salvaged

## Question 4

Bob dug a trench to divert water from a nearby stream to the survival camp. Alan dug a trench so waste could be safely flushed from the survival camp out to sea.

The two trenches need to be joined. When is the best time to integrate these two trenches?

- A. Integration should happen immediately assuming both trenches meet the definition of done
- B. Integration should be coordinated with the rest of the team during the Daily Scrum
- C. Integration should happen during the Sprint Review so stakeholders can be present
- D. Integration should happen after approval is obtained from the Scrum Master
- E. Integration should happen after approval is obtained from the Product Owner

## Question 5

Tina has just spotted a poisonous snake on the outskirts of the camp. When should Tina inform the rest of the development team?

- A. The development team should be informed immediately
- B. The development team should be informed at the next Daily Scrum
- C. The development team should be informed during the Sprint Retrospective
- D. The development team should be informed during the Sprint Review
- E. The development team should be informed after Tina talks about the deadly snake with the Scrum Master
- F. The development team should be informed after Tina talks about the deadly snake with the Product Owner

## Question 6

Living conditions have become dire and progress on the survival camp has not been going as quickly as planned. Two developers and the Scrum Master will be returning to the marooned ship for two days to care for loved ones.

What instructions do you, the Scrum Master, have for the developers returning to the boat, knowing that the survivors on the ship are the primary stakeholders in the development of the survival camp.

- A. Tell the developers not to directly interact with any of the stakeholders on the ship
- B. Tell the developers to direct all questions from stakeholders about the survival camp to the Product Owner
- C. Have developers tell stakeholders to hold any questions they have about the survival camp for the Sprint Review

- D. Discuss progress about the survival camp openly and honestly with stakeholders whenever they ask about it

## Question 7

Three healthy and enthusiastic survivors have swum ashore to help build the survival camp. Given the fact that a new, two-week Sprint has just started, when should these developers be added to the development team?

- A. The developers should be added to the team immediately
- B. The developers should be added to the team at the next Sprint Planning meeting
- C. The developers should be added to the team when the current Sprint ends but before the next Sprint begins
- D. The developers should be added to the team at the next Daily Scrum
- E. New developers cannot be added to a team once product development has started

## Question 8

Things have been running smoothly for the last three Sprints. However, three new developers with no experience in building a survival camp have just been added to the team.

What will happen to the individual productivity of the experienced developers?

- A. A permanent decrease in productivity will occur
- B. A short term decrease in productivity will occur
- C. A permanent increase in productivity will occur
- D. A short term increase in productivity will occur

## Question 9

The Product Owner has decided the survival camp needs a canoe. However, nobody on the team has the skills to build a canoe. What should the team do?

- A. Organize a sub-team dedicated to building a canoe
- B. Outsource the building of a canoe to a third-party
- C. Tell the Product Owner building a canoe is not possible
- D. Figure out how to build a canoe

## Question 10

A devastating flash flood has ripped through the survival camp and washed away the shelters the team spent the majority of the Sprint building.

How should the Scrum team proceed?

- A. Have the developers regroup and adjust the Sprint Plan accordingly
- B. Have the developers cancel the Sprint and begin a new Sprint Planning meeting
- C. Have the developers put time aside during the next Daily Scrum to adjust the Sprint Goal
- D. Tell the developers the Sprint Plan cannot be changed and they should continue with their original plan

Be @Scrumtuous and Follow @Xennial on Twitter

# Answers

## Answer 1

Scrum is software development process and should not be applied to life-or-death tasks like building a survival shelter.

- True
- False

This is false.

Scrum is a way to help teams get started on the development of a product and help those teams iteratively solve complex problems until product development is complete.

Scrum is a lightweight, incomplete framework for product development. Scrum is *not* a process or a methodology, and it is definitely not to exclusively tied to the domain of software development.

## Answer 2

You, the Scrum Master, have outdoors experience. You would like to be a developer on the first Sprint and help build the temporary shelter.

How should you proceed in order not to violate the rules of the Scrum Guide?

- A. You must temporarily assign Scrum Master duties to another person while you act as a developer
- B. You must temporarily assign Scrum Master duties to the Product Owner while you act as a developer
- C. You must participate in the Daily Scrum during the Sprint
- D. Scrum Masters cannot be a developer on their Scrum Team

Option C is correct.

There's no rule that says a Scrum Master can't also be a developer.

Could you imagine a life or death survival situation where a Scrum Master refuses to offer assistance just because they have the title of 'Scrum Master?' The idea is preposterous.

Both a Scrum Master and the Product Owner can participate as developers on a Scrum Team. The only rule is that when doing product development, a PO or Scrum Master must attend the Daily Scrum and participate as unbiased developers.

## Answer 3

A team of four developers, along with a Product Owner and the Scrum Master, have made it to shore and need to start building a survival shelter. However, the team is at odds over what they should do for the very first Sprint.

Whose approach is most in line with the ethos of the Scrum Guide?

- A. Jim wants to declare a 'Sprint 0' where time is spent planning and designing the survival camp
- B. Julie wants to spend the first Sprint searching the island for tools they can use to build the survival camp
- C. Jeff wants to spend the first Sprint searching to see if there is any edible food or clean water on the island
- D. Jessica wants to build a simple communication system from the camp to the ship using mirrors she salvaged

Option D is correct.

In Scrum, every Sprint must attempt to deliver some piece of usable and valuable functionality that will be part of the final product.

Setting up infrastructure, configuring tools, planning, designing and even looking to see if there are edible foods around are all noble tasks, but they all must occur in the background as the Scrum team focuses on the delivery of at least one Increment of value before the end of the current Sprint.

## Answer 4

Bob dug a trench to divert water from a nearby stream to the survival camp. Alan dug a trench so waste could be safely flushed from the survival camp out to sea.

The two trenches need to be joined. When is the best time to integrate these two trenches?

- A. Integration should happen immediately assuming both trenches meet the Definition of Done
- B. Integration should be coordinated with the rest of the team during the Daily Scrum
- C. Integration should happen during the Sprint Review so stakeholders can be present
- D. Integration should happen after approval is obtained from the Scrum Master
- E. Integration should happen after approval is obtained from the Product Owner

Option A is correct.

If two developers have worked on a feature that is complete and meets the Definition of Done, they should integrate their work immediately.

Scrum prides itself on being based on lean thinking. How lean would it be to have developers wait around for a Scrum Master or Product Owner to tell them it's okay to integrate their work?

If the work is done, it should be integrated immediately. After all, if the work does not successfully integrate, then it's not really 'done', is it?

Scrum doesn't put 'red tape', ceremony and procedure around the development process.

Instead, Scrum is pragmatic and lean. When work is done, integrate it immediately and then move on to the next thing that needs to be built.

## Answer 5

Tina has just spotted a poisonous snake on the outskirts of the camp. When should Tina inform the rest of the development team?

- A. The development team should be informed immediately
- B. The development team should be informed at the next Daily Scrum
- C. The development team should be informed during the Sprint Retrospective
- D. The development team should be informed during the Sprint Review
- E. The development team should be informed after Tina talks about the deadly snake with the Scrum Master
- F. The development team should be informed after Tina talks about the deadly snake with the Product Owner

Option A is correct.

Obviously if something threatens the people working on the product, the existence of the threat should be brought up immediately. You don't wait for a planned event to warn your peers about a dangerous situation.

On the Scrum Master exam, they will ask you what should happen if a developer identifies a dangerous security flaw that threatens the project. People often stumble on this question, thinking maybe it's something to discuss at the Sprint Retrospective, or something to raise at the next Daily Scrum.

If there's a serious problem, it should be addressed immediately. The Scrum Events should never act as an excuse to postpone important discussions about the health of the project.

## Answer 6

Living conditions have become dire and progress on the survival camp has not been going as quickly as planned. Two developers and the Scrum Master will be returning to the marooned ship for two days to care for loved ones.

What instructions do you, the Scrum Master, have for the developers returning to the boat, knowing that the survivors on the ship are the primary stakeholders in the development of the survival camp.

- A. Tell the developers not to directly interact with any of the stakeholders on the ship
- B. Tell the developers to direct all questions from stakeholders about the survival camp to the Product Owner
- C. Have developers tell stakeholders to hold any questions they have about the survival camp for the Sprint Review
- D. Discuss progress about the survival camp openly and honestly whenever stakeholders ask

Option D is correct.

Could you imagine how much fear, uncertainty and distrust it would create in this scenario if a gag order was put on the people building the survival camp?

Furthermore, the idea that developers should not be allowed to talk to stakeholders in this scenario is asinine and paradoxical, because every one of the developers is a stakeholder. Everyone in this problem domain is a stakeholder.

People think Scrum has a whole set of rules that regulate when a developer is allowed to talk to a stakeholder or the Product Owner or a customer. It doesn't.

Scrum doesn't advocate secrecy and silence. Scrum advocates for transparency and openness.

If your organization has some rules about developers not talking to stakeholders, that's an entirely different issue. It's also an impediment the Scrum Master might want to address.

Rules that limit transparency and communication do not coincide the Scrum values and pillars.

**Answer 7**

Three healthy and enthusiastic survivors have swum ashore to help build the survival camp. Given the fact that a new, two-week Sprint has just started, when should these developers be added to the development team?

- A. The developers should be added to the team immediately
- B. The developers should be added to the team at the next Sprint Planning meeting
- C. The developers should be added to the team when the current Sprint ends but before the next Sprint begins
- D. The developers should be added to the team at the next Daily Scrum
- E. New developers cannot be added to a team once product development has started

Option A is correct.

If you were in a life or death situation, and three people showed up who could potentially help save you, would you ask them to sit on the sidelines for two weeks until the next Sprint Planning meeting, or would you get them to work right away?

Which approach is more pragmatic and lean? Having developers sit on the sidelines doing nothing, or having them get to work right away? If new developers become available, have them join the team and get to work immediately.

**Answer 8**

Things have been running smoothly for the last three Sprints. However, three new developers with no experience in building a survival camp have just been added to the team.

What will happen to the individual productivity of the experienced developers?

- A. A permanent decrease in productivity will occur
- B. A short term decrease in productivity will occur
- C. A permanent increase in productivity will occur
- D. A short term increase in productivity will occur

Option B is correct.

When new developers are added to a team, the onboarding process takes time, and it usually consumes the time of existing team members.

When new developers become familiar with the development process, productivity will go back to normal, but in the short term, individual productivity will fall.

## Answer 9

The Product Owner has decided the survival camp needs a canoe. However, nobody on the team has the skills to build a canoe. What should the team do?

- A. Organize a sub-team dedicated to building a canoe
- B. Outsource the building of a canoe to a third-party
- C. Tell the Product Owner building a canoe is not possible
- D. Figure out how to build a canoe

Option D is correct.

It is assumed that all of the skills required to build out the various Product Backlog items exist within the team of Scrum developers, and if they don't, the team will acquire them.

There are no sub-teams or hierarchies in Scrum, and there's certainly nobody to outsource to on a deserted island.

When you take the Scrum Master certification exam, assume each test question is talking about an isolated team of developers unless the question states otherwise.

If one of the Product Backlog items is a canoe, then the team better learn how to carve out a canoe.

## Answer 10

A devastating flash flood has ripped through the survival camp and washed away the shelters the team spent the majority of the Sprint building.

How should the Scrum team proceed?

- A. Have the developers regroup and adjust the Sprint Plan accordingly
- B. Have the developers cancel the Sprint and begin a new Sprint Planning meeting
- C. Have the developers put time aside during the next Daily Scrum to adjust the Sprint Goal
- D. Tell the developers the Sprint Plan cannot be changed and they should continue with their original plan

Option A is correct.

When circumstances change, developers should adapt immediately.

Scrum provides a number of scheduled events to ensure communication happens regularly, but

those events are not the only times developers are allowed to communicate with each other.

If something dramatic happens, developers should immediately adapt. Adaptation should be perpetual throughout a Sprint.

Note that while the Sprint Plan can change, the Sprint Goal cannot. The Sprint Goal is finalized at the end of Sprint Planning and cannot be changed throughout the Sprint.

It is also not possible for the developers or the Scrum Master to cancel a Sprint. Only the Product Owner can cancel a Sprint, and only under the circumstance where the Sprint Goal has become obsolete.

Don't overthink Scrum. Don't ever let the Scrum events be an excuse for not taking immediate action when immediate action is warranted.

Be @Scrumtuous and Follow @Xennial on Twitter

## 25 Question Exam #1



**Good Luck!**

## Question 1

There are three roles, or as the Scrum Guide puts it, 'accountabilities' in Scrum:

- The Scrum Master
- The Product Owner
- The Developers

What are key characteristics and responsibilities of the Product Owner?

(Choose 2)

- A. The Product Owner is the one who orders the Product Backlog.
- B. The Product Owner is a single person and the role cannot be shared.
- C. The Product Owner commits 100% of their time to their Scrum Team.
- D. Multiple individuals with product knowledge can share the Product Owner role on a Scrum Team.

## Question 2

What are the primary objectives served by the Definition of Done (DoD)? (Choose 4)

- A. The DoD helps the Scrum Developers estimate the number of Product Backlog items to select for the Sprint.
- B. The DoD creates a common understanding across the team with regard to when development work is completed.
- C. The DoD describes the purpose, reasoning, and time allotment for each Scrum event.
- D. The DoD describes the work that must be done before the Sprint is ended.
- E. The DoD helps to increase transparency throughout the Scrum framework.
- F. The DoD helps to create consistency across Scrum Teams working on the same product.

## Question 3

Despite ongoing coaching and facilitation, a member of the development team has failed to integrate well with their peers.

The Scrum Developers have just informed you that the consensus is that the developer needs to be removed from the team.

What should you do as a Scrum Master? (Choose 1)

- A. Continue to facilitate team building activities to help the underperforming developer integrate better with the team.
- B. Continue to coach the development team on ways to improve communication between peers and help everyone on the team get along.
- C. Help the team remove the unwanted developer and coach them on ways to do it in a manner that is fair and respectful.
- D. Inform the development team that they do not have the authority to remove members from the team.

## Question 4

The product under development is very advanced and will likely be fairly complicated for new users.

Part of the Product Backlog includes the creation of extensive user documentation including screen-shots, how-to guides and even video demos.

According to the Scrum Guide, who is responsible for the delivery of technical documentation required for a software product?

(Choose 1)

- A. The Scrum Master creates all the technical documentation with oversight from the development team.
- B. Technical documentation is typically done by a specialized team external to the development team.
- C. Technical documentation is done by a specialized sub-team inside the Scrum Team.
- D. If it's a Product Backlog item, the Scrum Developers will need to find the time to create the technical documentation themselves.
- E. The creation of technical documentation is not a concern of a Scrum Team doing software development.

## Question 5

The Product Owner and the development team are at loggerheads over which Product Backlog items should be selected for the Sprint.

The Product Owner wants the highest priority items completed, but the development team insists existing impediments and unfulfilled dependencies make the highest priority backlog items impossible to start on. The development team has selected slightly lower priority items that can be completed this Sprint.

After 8 hours of Sprint Planning, the only thing the development team and the Product Owner have agreed upon is the Sprint Goal. What should the Scrum Master do? (Choose 3)

- A. Have the Product Owner choose the Sprint Backlog items for the development team to work on, as the Product Owner is the one who prioritized the Product Backlog.
- B. Trust the development team to choose which Product Backlog items to make part of the development Sprint.
- C. Extend the Sprint Planning meetings beyond its 8-hour timebox to allow the development team and the Product Owner to come to an agreement.
- D. End the Sprint Planning meeting and start development after the 8-hour Sprint Planning timebox ends, even if the disagreement persists.
- E. Discuss why the disagreement happened, how it impacted the development process and come up with ways to avoid such disagreements in the future during the Sprint Retrospective.

## Question 6

Every Sprint needs a Sprint Backlog. Which of the following two statements are true about the Sprint Backlog? (Choose 2)

- A. The Sprint Backlog is a list of all the work planned to be done in the Sprint.
- B. Every item has a specific developer designated as its owner.
- C. Each task is estimated in either hours or story points.
- D. The Sprint Backlog is managed by the Scrum developers.
- E. The Sprint Backlog is managed by the Scrum Master.
- F. The Sprint Backlog is managed by the Product Owner.

## Question 7

Which of the following would be responsibilities a team's Scrum Master would fulfill? (Choose 2)

- A. At the Sprint Review, the Scrum Master identifies which backlog items have met the Definition of Done and which have not.
- B. The Scrum Master coaches the Scrum Team on the importance of keeping Scrum events within their timebox.
- C. The Scrum Master helps to advise on and plan Scrum implementations within their organization.
- D. The Scrum Master will assign tasks to development team members in order to maximize their utilization.
- E. The Scrum Master is responsible for updating Agile tools such as JIRA and generating Sprint burn-down charts for the team.

## Question 8

In the absence of a person with quality assurance (QA) training on the team, what should a group of Scrum Developers do about all of the QA requirements that are part of the Definition of Done? (Choose 1)

- A. Have the organization form an external team of QA analysts who will take care of QA tasks.
- B. Complete each Product Backlog item and exclude QA tasks. Allow QA tasks to become part of a future Sprint.
- C. Have the development team train and learn the skills required to perform QA tasks.
- D. Have the Product Owner remove the quality assurance requirements from the Definition of Done.

## Question 9

While they are no longer officially part of the Scrum Guide, there are three questions commonly answered by developers in the Daily Scrum. What are they?

(Choose 3)

- A. How well do you feel the Scrum framework is being applied?
- B. What did you do yesterday?
- C. Will you be working overtime this weekend?
- D. What will you do today?
- E. What is the status you would like to report back to management?
- F. What, if anything, is blocking your progress?
- G. What Product Backlog items do you believe will need to be added to the next Sprint?

## Question 10

What techniques can a Scrum Master use to facilitate the prevention or removal of impediments in an organization?

(Choose 3)

- A. Coach management on how to use Scrum artifacts to achieve transparency.
- B. Schedule Zoom calls for the development team.
- C. Coach the Scrum Team on the importance of keeping the Sprint Retrospective within the timebox.
- D. Take notes for the developers during the Daily Scrum.
- E. Coach the developers on keeping the Daily Scrum within the timebox.

## Question 11

You have been hired to be the Scrum Master for five separate Scrum Teams, all of which are working to develop the same exact product.

As a Scrum Master intent on respecting the Scrum Guide and coaching around Scrum best practices, what would you recommend?

(Choose 2)

- A. There should be one Product Owner shared across all teams.
- B. There should be five Scrum Masters with each one 100% dedicated to their team.
- C. There should be five Product Backlogs, one for each team.
- D. There should be one Product Backlog shared across all teams.
- E. There should be five Product Owners with each one 100% dedicated to their team.
- F. Multiple Scrum Teams cannot develop the same product at the same time.

## Question 12

A developer has come to you, the Scrum Master, expressing concerns about the lax attitudes the team has towards data privacy and security. The concerned developer believes there are serious security vulnerabilities in the existing code.

How should the Scrum Master first respond to a Scrum Developer expressing such concerns?

(Choose 1)

- A. Ask the Product Owner to add security to the definition of done.
- B. Ask the development team to stop further development of features until the issues are addressed.
- C. Ask the Product Owner to add a Product Backlog item for security and push it to the top of the Product Backlog.
- D. Check with the organization's security team to see if the concerns are valid.
- E. Have the developer share their concerns with the team and have the team decide on a course of action.

## Question 13

The Scrum Team has been assembled, the Product Owner has the backlog ordered and everyone is ready for their very first Sprint!

What actions should a new Scrum Team take during the initial Sprint when starting a new project? (Choose 2)

- A. Create at least one piece of usable functionality that will be included in the final product.
- B. Perform requirements gathering and validate the initial designs so development can proceed in subsequent Sprints.
- C. With the Product Owner and Scrum Master, examine the Product Backlog and estimate how many Sprints will be required to complete the project.
- D. Communicate with external teams to understand how the continuous integration and continuous delivery pipelines work.
- E. Create a usable product Increment.

## Question 14

Human resources (HR) has just hired 75 new developers to work on a company-defining flagship product. The head of HR has asked you, the Scrum Master, to help with dividing these new hires into development teams.

What is the best way to divide a group of 75 new developers, of varying skills and experience, into multiple development teams?

(Choose 1)

- A. Nominate 7-10 trusted developers to be team leads and let the team leaders perform member allocation based on the skills their teams will require.
- B. The 75 developers should be allowed to divide themselves into teams on their own without direct intervention from the Scrum Master, Product Owner, stakeholders or outside party.
- C. Have the Scrum Master and Product Owner work together to create teams that balance skills, seniority and experience.
- D. Have an external, impartial party like the Human Resources department create teams that balance skills, seniority and experience.

## Question 15

The Sprint Goal is finalized during Sprint Planning and cannot be changed throughout the Sprint. (Choose 1)

- A. True
- B. False

## Question 16

The Sprint Review is one of Scrum's timeboxed events. According to the Scrum Guide, what is the maximum amount of time a Sprint Review should last? (Choose 1)

- A. 15 minutes
- B. 2 hours
- C. 4 hours
- D. 8 hours
- E. As long as the Scrum Team deems necessary.

## Question 17

What is the number one, highest priority and responsibility of the Product Owner in Scrum? (Choose 1)

- A. Working with the Scrum Master to select Product Backlog items to work on during the Sprint.
- B. Instilling quality by making sure developers adhere to their Definition of Done.
- C. Turning Product Backlog Items into understandable use cases and user stories.
- D. Maximizing the value of the product resulting from the work of the Scrum Team.

## Question 18

When three Scrum Teams work on the same product with the same Product Backlog, they must each have the same Sprint length. (Choose 1)

- True
- Flase

## Question 19

Two Scrum Teams working on the same product want to do Sprint Planning together so they can coordinate the items they pull from the common Product Backlog.

How would you as a Scrum Master advise the teams? (Choose 1)

- A. Two teams working on the same project should have separate Product Backlogs, making this unnecessary.
- B. Two separate Scrum Teams should not be openly discussing their development work with each other.
- C. The Product Owner should be selecting Product Backlog items for each team, so the joint meeting is unnecessary.
- D. Explain to the teams that Scrum forbids shared planning meetings.
- E. Encourage the teams to be self-managed, encourage them to introduce new processes like the one they suggested, and have them introspect during the Sprint Retrospective and adapt accordingly.

## Question 20

Sprint Zero is the name the Scrum Guide gives to the first Sprint in a project. (Choose 1)

- A. True
- B. False

## Question 21

Which of the following is true about Sprint Planning in Scrum? (Choose 2)

- A. The Sprint Goal must be finalized before the end of Sprint Planning and cannot change during the Sprint.
- B. The Sprint Backlog must be finalized before the end of Sprint Planning and cannot change during the Sprint.
- C. Scrum Teams working on the same Product Backlog may attend another team's Scrum Planning event.
- D. Scrum Teams working on the same Product Backlog may not attend another team's Scrum Planning event.

## Question 22

According to the 2020 Scrum Guide, when does a Sprint officially come to an end? (Choose 1)

- A. When the predetermined length of the Sprint expires.
- B. When all Product Backlog items are complete.
- C. When all Product Backlog items selected for the Sprint are complete.
- D. When a full increment that delivers value to the client is complete.

## Question 23

Ten separate Scrum teams are developing a very complex and highly integrated product.

What role does the Scrum Master play in ensuring each team can successfully integrate their software together with other teams without merge conflicts or incompatible deployments?

(Choose 1)

- A. The Scrum Master should have the Product Owner coordinate with a designated team lead from each Scrum Team and reach an agreement on when Product Backlog items will be integrated together.
- B. The Scrum Master should coach the ten teams on how to be self-managed and have the teams come up with their own solutions on how to integrate code together.
- C. The Scrum Master should visit each team at the start of their Sprints to ensure no two teams are working on the same task. This will make integration easier.
- D. Since the Product Owner (PO) is shared across all teams on the same project, the Scrum Master makes sure the PO visits each team regularly to coordinate integrations.

## Question 24

The Product Owner is not attending Sprint Reviews and Sprint Retrospectives. The Product Owner also seems to have trouble ordering the Product Backlog in a way that maximizes its value.

What should the Scrum Master do in this situation? (Choose 3)

- A. Coach the Product Owner and help them understand Scrum theory and practice.
- B. Report the issue to senior management.
- C. Discuss the issue with the Product Owner and try to cause the removal of impediments to the PO attending Scrum events.
- D. Coach the Product Owner on the importance of ordering the product backlog.
- E. Replace the Product Owner with a senior member of the development team.
- F. Remove the Product Owner from the Scrum Team.

## Question 25

Before the end of every Sprint, the Scrum Team must produce a usable, valuable Increment.

But what exactly constitutes a usable and valuable Increment?

How much work do the Scrum Developers need to perform on a Product Backlog item, or a decomposed Product Backlog item, for it to be considered a full-fledged Increment?

(Choose 1)

- A. The Scrum Developers do as much work as they can complete in the current Sprint, after which their work becomes part of the Increment.
- B. The Scrum Developers do a proportional amount of time on analysis, engineering, QA, programming, testing, and documentation.
- C. The Scrum Developers work on a Product Backlog item until the Scrum Master says it meets the Definition of Done
- D. The Scrum Developers work on a Product Backlog item until it meets the agreed upon Definition of Done.

## Answers

### Answer 1

There are three roles, or as the Scrum Guide puts it, 'accountabilities' in Scrum:

- The Scrum Master
- The Product Owner
- The Developers

What are key characteristics and responsibilities of the Product Owner?

- A. The Product Owner is the one who orders the Product Backlog.
- B. The Product Owner is a single person and the role cannot be shared.
- C. The Product Owner commits 100% of their time to their Scrum Team.
- D. Multiple individuals with product knowledge can share the PO role on a Scrum Team.

The correct answers are A and B.

The Product Owner is responsible for maximizing the value of the product resulting from the work of the Scrum Developers. The PO maintains the product backlog and ensures Product Backlog items are ordered.

The Product Owner is the champion of the project and the sole person responsible for managing the Product Backlog. They also ensure that the development team, stakeholders and others understand the product vision.

The Product Owner is the single point of accountability for the success of the product and is responsible for making decisions about the priority of work items. They ensure the work of the team delivers the most value to the stakeholders.

One of the reasons why only one person is allowed to do the Product Owner's job in Scrum is to ensure clear accountability and ownership. Having multiple people responsible for ordering the Product Backlog can lead to confusion and conflict about priorities and goals. In Scrum, the Product Owner is empowered to make product-related decisions and communicate them to the team. Having a single point of accountability helps to ensure that the team is aligned and focused on the same goals.

Additionally, having a single Product Owner helps to maintain consistency and coherence in the product vision and strategy. The Product Owner is responsible for articulating the product vision and ensuring that everyone on the team is working towards the same Product Goal.

Having a single person responsible for the Product Owner accountability in Scrum helps to ensure clear accountability, consistency in the product vision and effective communication with the team.

## Answer 2

What are the primary objectives served by the Definition of Done (DoD)?

- A. The DoD helps the Scrum Developers estimate the number of Product Backlog items to select for the Sprint.
- B. The DoD creates a common understanding across the team with regard to when development work is completed.
- C. The DoD describes the purpose, reasoning, and time allotment for each Scrum event.
- D. The DoD describes the work that must be done before the Sprint is ended.
- E. The DoD helps to increase transparency throughout the Scrum framework.
- F. The DoD helps to create consistency across Scrum Teams working on the same product.

The correct answers are A, B, E and F.

Scrum's "Definition of Done" is a shared understanding among the Scrum Team of what it means for a Product Backlog item to be considered complete. It is an agreed-upon set of criteria that a Product Backlog item (PBI), or a decomposition of a PBI, must meet before it can be considered "complete."

By establishing a clear Definition of Done, Scrum increases transparency by providing a shared understanding of what constitutes a completed Product Backlog item. This shared understanding helps ensure that everyone involved in the development process has the same understanding of what is expected in the final product.

The Definition of Done also helps developers estimate how much work to add to the Sprint Backlog. Because the team has a clear understanding of what constitutes a completed Product Backlog item, they can more accurately estimate how much work needs to be done to achieve that level of completeness. This can help the team better plan their work for the Sprint and avoid over-committing to work they may not be able to complete.

Finally, the DoD helps developers understand when work is complete. By providing a clear definition of what it means for a PBI to be considered complete, the team can avoid confusion or misunderstandings about what is expected from them. This clarity can help prevent issues such as incomplete work being considered done, leading to potential issues in the final product.

It's also stated in the Scrum Guide that when multiple teams work on the same Product, they should all share the same Definition of Done so they are all working within the same guidelines.

"If multiple Scrum Teams are working together on a product, they must mutually define and comply with the same Definition of Done." - The Scrum Guide, page 12.

## Answer 3

Despite ongoing coaching and facilitation, a member of the development team has failed to integrate well with their peers.

The Scrum Developers have just informed you that the consensus is that the developer needs to be removed from the team.

What should you do as a Scrum Master?

- A. Continue to facilitate team building activities to help the underperforming developer integrate better with the team.
- B. Continue to coach the development team on ways to improve communication between peers and help everyone on the team get along.
- C. Help the team remove the unwanted developer and coach them on ways to do it in a manner that is fair and respectful.
- D. Inform the development team that they do not have the authority to remove members from the team.

The correct answer is C.

Self-management is one of the core Scrum principles.

Self-management means that the team has the autonomy, authority and responsibility to manage themselves, including their own processes and procedures, without the need for external supervision or micromanagement.

One aspect of self-management is that the team members collectively hold themselves accountable for their performance and for delivering the work that they commit to completing within a given sprint. This means that if a team member is underperforming, it can impact the team's ability to meet their commitments and deliver on their goals.

In such situations, the concept of self-management gives the team the authority to take action and make decisions that are in the best interest of the team and the project. This includes the ability to remove an underperforming member from the team if necessary.

However, it is important to note that this decision should not be taken lightly and should be made collaboratively by the team as a whole. The Scrum Master can also provide guidance and support to the team in these situations to ensure that the decision is made fairly and with the best interests of the team and the project in mind.

## Answer 4

The product under development is very advanced and will likely be fairly complicated for new users.

Part of the Product Backlog includes the creation of extensive user documentation including screen-shots, how-to guides and even video demos.

According to the Scrum Guide, who is responsible for the delivery of technical documentation required for a software product?

- A. The Scrum Master creates all the technical documentation with oversight from the development team.
- B. Technical documentation is typically done by a specialized team external to the development team.
- C. Technical documentation is done by a specialized sub-team inside the Scrum Team.
- D. If it's a Product Backlog item, the Scrum Developers will need to find the time to create the technical documentation themselves.
- E. The creation of technical documentation is not a concern of a Scrum Team doing software development.

The correct answer is D.

"Developers are the people in the Scrum Team that are committed to creating any aspect of a usable Increment each Sprint."

A Scrum Team is responsible for the creation of an increment that meets the Definition of Done, and it is the developers who do the work. If technical documentation is a Product Backlog item, or it is included in the Definition of Done, the developers on the Scrum Team are expected to have the skills and time to create it.

"Scrum Teams are cross-functional, meaning the members have all the skills necessary to create value each Sprint."

Furthermore, there are no 'special teams' in Scrum. The Scrum Guide clearly states: "Within a Scrum Team, there are no sub-teams or hierarchies."

In Scrum, the Scrum Developers are responsible for creating a potentially releasable product Increment every Sprint that meets the Definition of Done. If user documentation is part of the "Done" criteria, then the team of developers must find a way to produce it.

"The Scrum Team is responsible for all product-related activities from stakeholder collaboration, verification, maintenance, operation, experimentation, research and development and anything else that might be required."

## Answer 5

The Product Owner and the development team are at loggerheads over which Product Backlog items should be selected for the Sprint.

The Product Owner wants the highest priority items completed, but the development team insists existing impediments and unfulfilled dependencies make the highest priority backlog items impossible to start on. The development team has selected slightly lower priority items that can be completed this Sprint.

After 8 hours of Sprint Planning, the only thing the development team and the Product Owner have agreed upon is the Sprint Goal. What should the Scrum Master do?

- A. Have the Product Owner choose the Sprint Backlog items for the development team to work on, as the Product Owner is the one who prioritized the Product Backlog.
- B. Trust the development team to choose which Product Backlog items to make part of the development Sprint.
- C. Extend the Sprint Planning meetings beyond its 8-hour timebox to allow the development team and the Product Owner to come to an agreement.
- D. End the Sprint Planning meeting and start development after the 8-hour Sprint Planning timebox ends, even if the disagreement persists.
- E. Discuss why the disagreement happened, how it impacted the development process and come up with ways to avoid such disagreements in the future during the Sprint Retrospective.

The correct answers are B, D and E.

You don't ever extend a Scrum event beyond its timebox.

Disagreements will happen. When they do, move forward and start development.

A Sprint will only last a month at most. And teams can adapt daily during the Daily Scrum as the situation changes. That includes potentially adding new PBIs or removing PBIs from the Sprint backlog.

The development team must be trusted to choose what is right in terms of what to develop over the short term.

**Answer 6**

Every Sprint needs a Sprint Backlog. Which of the following two statements are true about the Sprint Backlog?

- A. The Sprint Backlog is a list of all the work planned to be done in the Sprint.
- B. Every item has a specific developer designated as its owner.
- C. Each task is estimated in either hours or story points.
- D. The Sprint Backlog is managed by the Scrum developers.
- E. The Sprint Backlog is managed by the Scrum Master.
- F. The Sprint Backlog is managed by the Product Owner.

The correct answers are A and D.

The Sprint Backlog is a key artifact in the Scrum framework.

The Sprint Backlog outlines the work that the Scrum Developers plan to complete during the current Sprint. It's created and managed by the developers, which makes sense because it is the developers who are responsible for delivering at least one Increment before the end of the Sprint.

The Sprint Backlog is managed by the Scrum Developers because they are the ones who are responsible for organizing, planning, and executing the work. The developers have the technical knowledge required to create the plan, estimate the work and track the progress during the Sprint.

You can always assume that the Product Owner has a vision, but has absolutely no clue how to build it. Only the developers have that knowledge.

**Answer 7**

Which of the following would be responsibilities a team's Scrum Master would fulfill?

- A. At the Sprint Review, the Scrum Master identifies which backlog items have met the Definition of Done and which have not.
- B. The Scrum Master coaches the Scrum Team on the importance of keeping Scrum events within their timebox.
- C. The Scrum Master helps to advise on and plan Scrum implementations within their organization.
- D. The Scrum Master will assign tasks to development team members in order to maximize their utilization.
- E. The Scrum Master is responsible for updating Agile tools such as JIRA and generating Sprint burn-down charts for the team.

The correct answers are B and C.

The Scrum Master is responsible for ensuring the Scrum framework is followed by the team and that the team is continuously improving. Two key aspects of this responsibility are facilitating timeboxed meetings and coaching people on Scrum values.

The Scrum Master is also responsible for helping their organization to plan for Scrum adoption and advise on how to implement Scrum successfully.

From the Scrum Guide:

The Scrum Master serves the organization in several ways, including:

- Leading, training, and coaching the organization in its Scrum adoption;
- Planning and advising Scrum implementations within the organization;
- Helping employees and stakeholders understand and enact an empirical approach for complex work; and,
- Removing barriers between stakeholders and Scrum Teams.

## Answer 8

In the absence of a person with quality assurance (QA) training on the team, what should a group of Scrum Developers do about all of the QA requirements that are part of the Definition of Done?

- A. Have the organization form an external team of QA analysts who will take care of QA tasks.
- B. Complete each Product Backlog item and exclude QA tasks. Allow QA tasks to become part of a future Sprint.
- C. Have the development team train and learn the skills required to perform QA tasks.
- D. Have the Product Owner remove the quality assurance requirements from the Definition of Done.

The correct answer is C.

The Scrum development team must be cross-functional and capable of fulfilling all of the requirements that make up the Definition of Done. If any of those skills are lacking, the team will need to acquire those skills.

In this case, if QA is required to meet the Definition of Done, then someone, or everyone on the development team, must learn QA.

## Answer 9

While they are no longer officially part of the Scrum Guide, there are three questions commonly answered by developers in the Daily Scrum. What are they?

- A. How well do you feel the Scrum framework is being applied?
- B. What did you do yesterday?
- C. Will you be working overtime this weekend?
- D. What will you do today?
- E. What is the status you would like to report back to management?
- F. What, if anything, is blocking your progress?
- G. What Product Backlog items do you believe will need to be added to the next Sprint?

The correct answers are B, D and F.

At the Daily Scrum, the Scrum Developers commonly share what was accomplished since the last Daily Scrum, any impediments they are facing, and what they plan to do between now and the next Daily Scrum.

Also known as the '3 Daily Scrum questions', these questions were included in previous version of the Scrum Guide, but were removed in the 2020 Scrum Guide update, not because they are bad, but because developers should be free to run the Daily Scrum in any manner they like.

## Answer 10

What techniques can a Scrum Master use to facilitate the prevention or removal of impediments in an organization?

- A. Coach management on how to use Scrum artifacts to achieve transparency.
- B. Schedule Zoom calls for the development team.
- C. Coach the Scrum Team on the importance of keeping the Sprint Retrospective within the timebox.
- D. Take notes for the developers during the Daily Scrum.
- E. Coach the developers on keeping the Daily Scrum within the timebox.

The correct answers are A, C and E.

One of the responsibilities of the Scrum Master is to coach the organization in the proper application of Scrum. Another responsibility is to cause the removal of impediments from the team.

When it comes to causing the removal of impediments, those impediments deal specifically with issues related to Scrum and the inappropriate application of Scrum.

If an impediment is that a team is being micromanaged, the Scrum Master would cause the removal of that impediment by coaching management on the importance of self-management and respecting the autonomy of the Scrum Development team.

For the Scrum Team itself, the Scrum Master's job is to coach them on the proper application of the Scrum Guide.

The Scrum Guide doesn't have a lot of rules, but some of the rules that it does have revolve around how long meetings can last. Coaching the team on how to keep Scrum Events within their timebox is definitely a responsibility of a Scrum Master, and if meetings are going too long, helping to keep them short will help cause the removal of that impediment.

"The Scrum Master serves the Scrum Team in several ways, including causing the removal of impediments to the Scrum Team's progress."

## Answer 11

You have been hired to be the Scrum Master for five separate Scrum Teams, all of which are working to develop the same exact product.

As a Scrum Master intent on respecting the Scrum Guide and coaching around Scrum best practices, what would you recommend?

- A. There should be one Product Owner shared across all teams.
- B. There should be five Scrum Masters with each one 100% dedicated to their team.
- C. There should be five Product Backlogs, one for each team.
- D. There should be one Product Backlog shared across all teams.
- E. There should be five Product Owners with each one 100% dedicated to their team.
- F. Multiple Scrum Teams cannot develop the same product at the same time.

The correct answers are A and D.

In Scrum, the Product Backlog is the single, ordered list of items that define what needs to be done to complete the project. The Product Owner is responsible for managing the Product Backlog, ensuring that it is constantly refined and ordered, and that the development team has a clear understanding of what needs to be built and why.

When multiple teams are building the same product, they must share:

- A common Product Backlog

- The same Product Owner
- A common Product Goal
- The same Definition of Done

## Answer 12

A developer has come to you, the Scrum Master, expressing concerns about the lax attitudes the team has towards data privacy and security. The concerned developer believes there are serious security vulnerabilities in the existing code.

How should the Scrum Master first respond to a Scrum Developer expressing such concerns?

- A. Ask the Product Owner to add security to the definition of done.
- B. Ask the development team to stop further development of features until the issues are addressed.
- C. Ask the Product Owner to add a Product Backlog item for security and push it to the top of the Product Backlog.
- D. Check with the organization's security team to see if the concerns are valid.
- E. Have the developer share their concerns with the team and have the team decide on a course of action.

The correct answer is E.

A Scrum Developer should discuss major concerns immediately with their development team rather than waiting for a Sprint Review or Sprint Retrospective to do so.

If the developer does not feel comfortable discussing the issue with the team, that points to an even deeper problem with team dynamics, which is an impediment to transparency. This would be an impediment the Scrum Master would be interested in coaching around.

## Answer 13

The Scrum Team has been assembled, the Product Owner has the backlog ordered and everyone is ready for their very first Sprint!

What actions should a new Scrum Team take during the initial Sprint when starting a new project?

- A. Create at least one piece of usable functionality that will be included in the final product.
- B. Perform requirements gathering and validate the initial designs so development can proceed in subsequent Sprints.
- C. With the Product Owner and Scrum Master, examine the Product Backlog and estimate how many Sprints will be required to complete the project.
- D. Communicate with external teams to understand how the continuous integration and continuous delivery pipelines work.
- E. Create a usable product Increment.

The correct answers are A and E.

Every Sprint in Scrum must produce a valuable, usable Increment of work and this Increment of work must be something that becomes part of the final product.

There is no Sprint Zero in Scrum. There is no 'initial setup Sprint' in Scrum where teams can focus on designs or infrastructure.

Every Sprint must produce something of value. Every Sprint must produce an Increment of work that will incrementally, over time, lead to a complete and finished product.

## Answer 14

Human resources (HR) has just hired 75 new developers to work on a company-defining flagship product.

The head of HR has asked you, the Scrum Master, to help with dividing these new hires into development teams.

What is the best way to divide a group of 75 new developers, of varying skills and experience, into multiple development teams?

- A. Nominate 7-10 trusted developers to be team leads and let the team leaders perform member allocation based on the skills their teams will require.
- B. The 75 developers should be allowed to divide themselves into teams on their own without direct intervention from the Scrum Master, Product Owner, stakeholders or outside party.
- C. Have the Scrum Master and Product Owner work together to create teams that balance skills, seniority and experience.
- D. Have an external, impartial party like the Human Resources department create teams that balance skills, seniority and experience.

The correct answer is B.

Dividing a group of 75 developers into multiple development teams can be a complex process that requires careful consideration of many factors including the developers' skills, experience and the nature of the project.

One of the best and most Agile ways to divide a group is to use a self-organizing approach, which allows the developers to organize themselves into teams based on their skills and experience. This approach has several benefits:

- Empowerment: When developers are allowed to organize themselves, they feel empowered and are more likely to take ownership of their work and be motivated to succeed.
- Autonomy: Self-organizing teams have the autonomy to make decisions about how to approach their work, which can lead to more innovative solutions and increased productivity.
- Flexibility: The self-organizing approach allows for flexibility in team composition and size, which can be adjusted as the project evolves.

To implement this approach, the company can provide guidelines and support for the developers to form their teams. Giving the developers insight into the product, what is being built and the skill required would be helpful as well. Informed decisions are the best decisions, but there should not be direct intervention from any outside parties.

This also comes with the understanding that the teams will be allowed to re-organize again in the future if the teams believe it is necessary to do so.

## Answer 15

The Sprint Goal is finalized during Sprint Planning and cannot be changed throughout the Sprint.

- A. True  
 B. False

The correct answer is A.

This is true.

The Sprint Goal is finalized during Sprint Planning and remains constant throughout the Sprint.

Changes can be made to the Sprint Plan and the Sprint Backlog during a Sprint, but the Sprint Goal remains constant.

"The whole Scrum Team then collaborates to define a Sprint Goal that communicates why the Sprint is valuable to stakeholders. The Sprint Goal must be finalized prior to the end of Sprint Planning."

## Answer 16

The Sprint Review is one of Scrum's timeboxed events. According to the Scrum Guide, what is the maximum amount of time a Sprint Review should last?

- A. 15 minutes
- B. 2 hours
- C. 4 hours
- D. 8 hours
- E. As long as the Scrum Team deems necessary.

The correct answer is C.

The Sprint Review should only last for 4 hours.

You must know the maximum allowed timebox for each event in Scrum:

- 15 minutes for the Daily Scrum
- 3 hours for the Sprint Retrospective
- 4 hours for the Sprint Review
- 8 hours for Sprint Planning
- One calendar month for a Sprint

From the Scrum Guide: "The Sprint Review is the second to last event of the Sprint and is timeboxed to a maximum of four hours for a one-month Sprint. For shorter Sprints, the event is usually shorter."

## Answer 17

What is the number one, highest priority and responsibility of the Product Owner in Scrum?

- A. Working with the Scrum Master to select Product Backlog items to work on during the Sprint.
- B. Instilling quality by making sure developers adhere to their Definition of Done.
- C. Turning Product Backlog Items into understandable use cases and user stories.
- D. Maximizing the value of the product resulting from the work of the Scrum Team.

The correct answer is D.

The answer to this question comes right out of the Scrum Guide:

"The Product Owner is accountable for maximizing the value of the product resulting from the work of the Scrum Team."

The Scrum Guide does not mention the terms "use cases" or "user stories," so any references to them on the Scrum certification exam is normally an incorrect option.

"Instilling quality by adhering to a Definition of Done" is actually something the Developers are accountable for according to the Scrum Guide, not the Product Owner.

Note that the *developers* select Product Backlog Items to work on during the Sprint, not the Scrum Master or the Product Owner.

## Answer 18

When three Scrum Teams work on the same product with the same Product Backlog, they must each have the same Sprint length.

- True
- False

The correct answer is B.

There is nothing in the Scrum Guide that requires teams working on the same product to have the same Sprint length.

If the Scrum Guide does not directly address a topic, then there are no specific rules about it.

The Scrum Guide says teams working on the same product must have:

- The same Product Owner
- The same Product Goal
- The same Product Backlog
- The same Definition of Done

Should the teams start their Sprints on the same day? End Sprints on the same day? Make their Sprints the same length?

The Scrum Guide is silent on the issue. It's up to the team to decide what is best.

Scrum self-identifies as an 'incomplete framework'. It's just a guide. It doesn't have all the answers.

## Answer 19

Two Scrum Teams working on the same product want to do Sprint Planning together so they can coordinate the items they pull from the common Product Backlog.

How would you as a Scrum Master advise the teams?

- A. Two teams working on the same project should have separate Product Backlogs, making this unnecessary.
- B. Two separate Scrum Teams should not be openly discussing their development work with each other.
- C. The Product Owner should be selecting Product Backlog items for each team, so the joint meeting is unnecessary.
- D. Explain to the teams that Scrum forbids shared planning meetings.
- E. Encourage the teams to be self-managed, encourage them to introduce new processes like the one they suggested, and have them introspect during the Sprint Retrospective and adapt accordingly.

The correct answer is E.

Scrum is self-described as an incomplete framework. That means there is a great deal of flexibility and creativity allowed within the bounds of what constitutes Scrum.

Generally speaking, if Scrum doesn't explicitly forbid it, then teams are free to explore it.

Shared Sprint Planning meetings might be a great process for the teams to implement. Many large companies do exactly that. The Scrum Master should encourage such experimentation.

## Answer 20

Sprint Zero is the name the Scrum Guide gives to the first Sprint in a project.

- A. True
- B. False

The correct answer is B.

There is no such thing as a 'Sprint Zero' according to the Scrum Guide.

Furthermore, the Scrum Guide makes no references to a 'Release Sprint,' 'Integration Sprint' or an 'Infrastructure Sprint.'

These phrases on the Scrum Master certification exam will almost always be red herrings that

point to an incorrect answer.

Scrum just has Sprints, plain ordinary Sprints, that must create a usable and valuable Increment of work.

There are no Sprints in Scrum where you 'just set things up' or 'just get the infrastructure working.'

There is no such thing as a 'Sprint Zero' in Scrum.

## Answer 21

Which of the following is true about Sprint Planning in Scrum?

- A. The Sprint Goal must be finalized before the end of Sprint Planning and cannot change during the Sprint.
- B. The Sprint Backlog must be finalized before the end of Sprint Planning and cannot change during the Sprint.
- C. Scrum Teams working on the same Product Backlog may attend another team's Scrum Planning event.
- D. Scrum Teams working on the same Product Backlog may not attend another team's Scrum Planning event.

The correct answers are A and C.

The Sprint Goal must be finalized before the end of Sprint Planning, but not the Sprint Backlog. The Sprint Backlog is allowed to change throughout the Sprint.

Also, anyone can attend Sprint Planning if they are invited, so other teams can come and help with the selection of PBIs. This can be very helpful if multiple teams are working on the same project.

## Answer 22

According to the 2020 Scrum Guide, when does a Sprint officially come to an end?

- A. When the predetermined length of the Sprint expires.
- B. When all Product Backlog items are complete.
- C. When all Product Backlog items selected for the Sprint are complete.
- D. When a full increment that delivers value to the client is complete.

The correct answer is A.

A Sprint expires when the predetermined length for the Sprint expires.

A Sprint may be two weeks, three weeks, four weeks or a full calendar month. That's up to the team to decide.

Once the time allotted for the Sprint expires, the Sprint ends and a new Sprint begins.

Even if Product Backlog items are not finished, or even if the team failed to deliver any Increment of value, the Sprint is not extended.

To answer this question even more concisely, the Sprint ends at the end of the Sprint Retrospective. The Sprint Retrospective is the last event in Scrum, and when this event finishes, a new Sprint begins.

## Answer 23

Ten separate Scrum teams are developing a very complex and highly integrated product.

What role does the Scrum Master play in ensuring each team can successfully integrate their software together with other teams without merge conflicts or incompatible deployments?

- A. The Scrum Master should have the Product Owner coordinate with a designated team lead from each Scrum Team and reach an agreement on when Product Backlog items will be integrated together.
- B. The Scrum Master should coach the ten teams on how to be self-managed and have the teams come up with their own solutions on how to integrate code together.
- C. The Scrum Master should visit each team at the start of their Sprints to ensure no two teams are working on the same task. This will make integration easier.
- D. Since the Product Owner (PO) is shared across all teams on the same project, the Scrum Master makes sure the PO visits each team regularly to coordinate integrations.

The correct answer is B.

If you think a Scrum Master or a Product Owner is going to be able to figure out how to integrate code across 10 separate teams working on a complicated project, they you've never met a Scrum Master or a Product Owner.

The Product Owner manages the Product Backlog and the Scrum Master coaches and facilitates, but they can't solve technical issues for the development teams.

All a Scrum Master can do is coach those teams and help them discover their own solutions that work within the intentionally incomplete framework known as Scrum.

## Answer 24

The Product Owner is not attending Sprint Reviews and Sprint Retrospectives.

The Product Owner also seems to have trouble ordering the Product Backlog in a way that maximizes its value.

What should the Scrum Master do in this situation?

- A. Coach the Product Owner and help them understand Scrum theory and practice.
- B. Report the issue to senior management.
- C. Discuss the issue with the Product Owner and try to cause the removal of impediments to the PO attending Scrum events.
- D. Coach the Product Owner on the importance of ordering the product backlog.
- E. Replace the Product Owner with a senior member of the development team.
- F. Remove the Product Owner from the Scrum Team.

The correct answers are A, C and D.

It is the Scrum Master's job to work with team members and help them understand Scrum theory and practice.

Coaching the Product Owner on why Scrum events are important and facilitating ways the Product Owner can get more value out of these Scrum events is one of the Scrum Master's jobs.

The Scrum Master should also coach this Product Owner on the fact that the PO is the one who maximizes the value generated by the Scrum Team, and the primary way that happens is by properly ordering the Product Backlog.

"The Scrum Master is accountable for establishing Scrum as defined in the Scrum Guide. They do this by helping everyone understand Scrum theory and practice, both within the Scrum Team and the organization."

## Answer 25

Before the end of every Sprint, the Scrum Team must produce a usable, valuable Increment.

But what exactly constitutes a usable and valuable Increment?

How much work do the Scrum Developers need to perform on a Product Backlog item, or a decomposed Product Backlog item, for it to be considered a full-fledged Increment?

- A. The Scrum Developers do as much work as they can complete in the current Sprint, after which their work becomes part of the Increment.
- B. The Scrum Developers do a proportional amount of time on analysis, engineering, QA, programming, testing, and documentation.
- C. The Scrum Developers work on a Product Backlog item until the Scrum Master says it meets the Definition of Done
- D. The Scrum Developers work on a Product Backlog item until it meets the agreed upon Definition of Done.

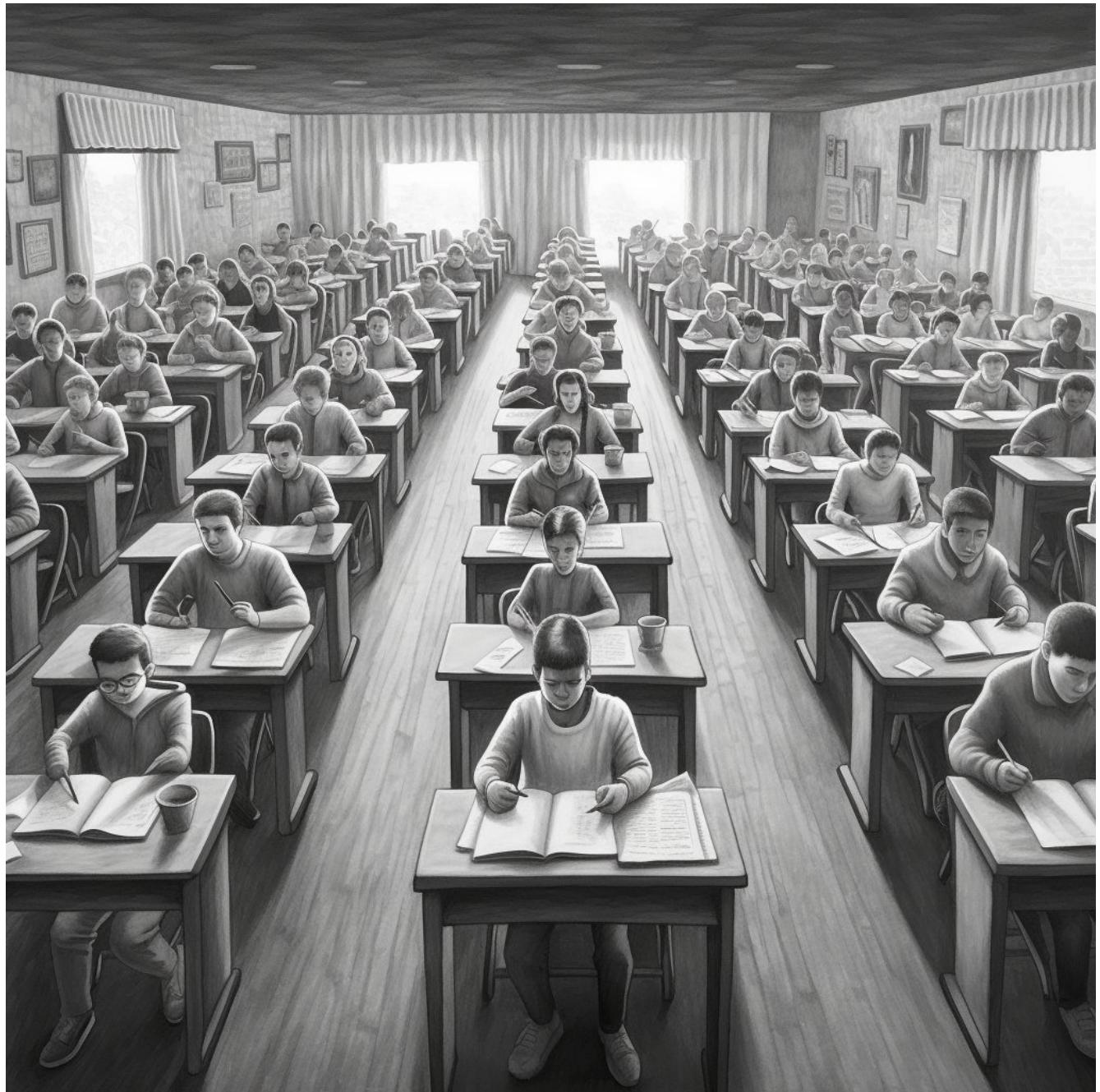
The correct answer is D.

During Sprint Planning, the Scrum Developers pick Product Backlog items they believe they can complete before the end of the Sprint.

When a selected item meets the Definition of Done, the feature is said to be complete, the work becomes part of the Increment, and the results are shared with stakeholders at the Sprint Review.

If developers fail to complete the Product Backlog item during the Sprint, the item returns to the Product Backlog. It may or may not be selected for the next Sprint. That's up to the team to decide at the next Sprint planning event.

## 25 Question Exam #2



**Good Luck!**

## Question 1

The Sprint Planning meeting is one of Scrum's timeboxed events.

What is the timebox for the Sprint Planning meeting? (Choose 1)

- A. A minimum of 4 hours
- B. A maximum of 4 hours
- C. A minimum of 8 hours
- D. A maximum of 8 hours

## Question 2

The Scrum Guide demands that management allows teams to be self-managed.

What advantages does self-management offer to Scrum Teams and the team's members? (Choose 3)

- A. Increased personal accountability with regards to the work developers perform.
- B. Increased accuracy in estimating Product Backlog Item completion dates.
- C. Increased personal commitment to the team and the project.
- D. Increased ingenuity and inventiveness when solving problems.
- E. Increased ability to comply with industry regulation, rules and guidelines.

## Question 3

What should a Scrum Master do if the Scrum Developers are missing the necessary skills, tools and infrastructure needed to complete the highest-priority backlog items? (Choose 2)

- A. Dissolve the Sprint and inform management that the Scrum Team is not ready to use the Scrum framework.
- B. Instruct the Product Owner to accept partially Done Increments for this Sprint.
- C. Change the priority of the current Sprint to improving the development team's skills and acquiring the necessary infrastructure instead of delivering an Increment.
- D. Coach the Scrum Developers on how to spend time during the Sprint learning and improving their skills and acquiring tools they will need to complete important Backlog Items.
- E. Discuss adjusting the Definition of Done to address the tooling issue.

## Question 4

When should newly identified work or additional information about work be added to the Sprint Backlog? (Choose 1)

- A. New insight about the Sprint Plan should be added to the Sprint Backlog immediately.
- B. When the Product Owner agrees that the information should be added to the Sprint Backlog
- C. Sprint Backlog items can only be added during Sprint Planning.
- D. During the Daily Scrum on approval from the team lead

## Question 5

In Scrum, testers, QA personnel, and business analysts on a Scrum Team are considered developers. (Choose 1)

- A. True
- B. False

## Question 6

What effects would be observed on the original Scrum Team when two additional Scrum Teams are included in the development process for the same product? (Choose 1)

- A. The original Scrum team's productivity is likely to decrease slightly.
- B. The original Scrum team's productivity is likely to stay the same.
- C. The original Scrum team's productivity is likely to increase slightly.
- C. The original Scrum team's productivity is likely to increase significantly.

## Question 7

What should the Scrum Team do with a Product Backlog item that was added to the Sprint Backlog if it does not meet the Definition of Done at the end of a Sprint? (Choose 2)

- A. The Product Owner can release it with the permission of the stakeholders.
- B. Return it to the Product Backlog and let the Product Owner reorder it.
- C. Award partial points for the work done so velocity is recorded and return what remains to the Product Backlog.
- D. Do not present the progress made on the incomplete item to the Stakeholders or customers

## Question 8

During Sprint Planning, Scrum Developers choose the various Product Backlog items they believe they can transform into a usable and valuable Increment before the Sprint ends.

What level of effort and resolve must Scrum developers put into completing the Sprint Backlog items they have chosen for the Sprint? (Choose 1)

- A. The developers will work at a sustainable pace in order to complete as many Sprint Backlog items as possible in conformance with the Definition of Done.
- B. The developers must put in all the effort required to successfully complete all Sprint Backlog items before the Sprint is complete.
- C. The developers must commit to delivering core functionality but testing and quality assurance is not required.
- D. The developers must complete the selected Backlog Items in accordance with the Definition of Done, and also provide unit testing, technical documentation and also pass any QA routines.

## Question 9

Who is best equipped to estimate the amount of time required to complete a Product Backlog item(PBI)? (Choose 1)

- A. The Product Owner, after adding the PBI to the Product Backlog.
- B. The Scrum Master, after reading the description of the PBI.
- C. The Project Manager, after assigning the PBI to the project.
- D. The individual developer tasked with working on the PBI.
- E. The team of Scrum Developers, after discussions with the Product Owner about the PBI.

## Question 10

A feedback loop is an opportunity for a team or group to discuss, inspect and most importantly, adapt.

Which of these events provide feedback loops in Scrum? (Choose 3)

- A. The Sprint Retrospective
- B. The Daily Standup
- C. The Sprint Review
- D. The Performance Review
- E. The Daily Scrum

## Question 11

The Scrum Master sees the Product Owner (PO) struggle with the task of ordering the Product Backlog. What action would a certified Scrum Master take in these circumstances? (Choose 1)

- A. Have the Product Owner extend the length of the current Sprint to have more time to order the Product Backlog before the next Sprint begins.
- B. Inform the Product owner that it is the development team's responsibility to order the Product Backlog.
- C. Coach the Product Owner on the importance of ordering the Product Backlog and how the PO is responsible for an order that will deliver the greatest value.
- D. The Scrum Master should provide the Product Owner with the Product Backlog that was ordered by the Scrum Development Team with the Scrum Master's oversight and guidance.
- E. Tell the Product Owner to work with the developers to prioritize Backlog Items based on which items will be the easiest and fastest to implement.

## Question 12

The DevOps team is putting software into production that fails in terms of performance and security. The development team says they only deal with functional requirements, not non-functional requirements like security and performance.

What's the best way for the Scrum Master to deal with this? (Choose 2)

- A. Share stories from the help desk about the issues the deployed software is having.
- B. Make sure non-functional requirements are clearly articulated in the Definition of Done.
- C. Run continuous integration tests throughout the Sprint and have developers address performance and security issues that arise before deployment.
- D. Ask the Product Owner to add new Product Backlog items that deal specifically with performance and security.
- E. As the Scrum Master, you should push back on the DevOps team and explain that non-functional requirements are not part of the development team's planning tasks.

## Question 13

Sprints are kept to 30 days or less because: (Choose 3)

- A. Risk increases with longer Sprints.
- B. Shorter Sprints are easier to budget.
- C. Complexity is likely to rise with longer Sprints.
- D. Monthly Sprints integrate easier with reporting and tracking tools like JIRA
- E. The likelihood that the Sprint Goal will become invalid increases with longer Sprints.

## Question 14

According to the Scrum Guide, the Sprint Backlog is composed of three interdependent parts.

Which three of the following items together make up the Sprint Backlog? (Choose 3)

- A. The Sprint Goal
- B. The Product Goal
- C. An actionable plan
- D. Product backlog items chosen for the Sprint
- E. Sprint backlog items not completed on the previous Sprint

## Question 15

Who is responsible for removing barriers between stakeholders and Scrum Teams? (Choose 1)

- A. The Product Owner
- B. The Scrum Master
- C. The Project Manager
- D. The Development Team

## Question 16

Sometimes a team has too many developers. Other times a developer just might not be the right fit for a development team.

Who has the power to remove an underperforming developer from a Scrum team? (Choose 1)

- A. Only someone in management or Human Resources can remove an underperforming developer from a Scrum team. This is not a Scrum responsibility.
- B. The Scrum Master can remove underperforming developers from a Scrum team.
- C. The responsibility to remove a developer from the Scrum Team lies outside of the scope of the Scrum Guide.
- D. The team of Scrum Developers has the final say over who is allowed to be a member of the development team.

## Question 17

Which of the following statements is true about project and Sprint cancellations? (Choose 1)

- A. Only the Scrum Master can cancel a Sprint.
- B. Only the Product Owner can cancel a Sprint.
- C. Only the Scrum Master can cancel the project.
- D. Only the Product Owner can cancel the project.

## Question 18

According to the Scrum Guide, what is the purpose of the Sprint Backlog? (Choose 2)

- A. The Sprint Backlog is designed to help user stories get estimated in story points.
- B. The Sprint Backlog is a list of all work the team hopes to complete in the current Sprint.
- C. The Sprint Backlog is a list of all work the team hopes to complete in the next Sprint.
- D. The Sprint Backlog is the Scrum Developers' actionable plan for the Sprint.
- E. The Sprint Backlog is ordered and arranged by the Product Owner to maximize the developer's productivity.

## Question 19

Which one of the following is the most accurate description of Scrum as defined by the Scrum Guide?

- A. A complete, incremental, and iterative software development process
- B. An incomplete, incremental, and iterative software development process
- C. A purposefully incomplete, lightweight framework to help find solutions for complex problems
- D. A purposefully complete, lightweight framework to help find solutions for complex problems

## Question 20

Three Scrum Teams are working on the same product. Each team produces an increment of work by the end of their Sprint. When should the Increment from a given team be integrated into the product? (Choose 1)

- A. Teams should be doing continuous integration, and increments must be integrated with the work of other teams before the Sprint review.
- B. Teams should be doing continuous integration, and increments must be integrated with the work of other teams after the Sprint review.
- C. Three Scrum teams working on the same project should be independent of each other and

not integrate their work.

- D. The three teams should only integrate their code at the end of the project.

## Question 21

Who holds developers accountable for creating a plan, instilling quality, and adhering to the Definition of Done? (Choose 1)

- A. The Product Owner holds the developers accountable
- B. The Scrum Master holds the developers accountable
- C. The Stakeholders hold the developers accountable
- D. Developers hold each other accountable as professionals

## Question 22

What is the best way for the Product Owner to ensure the Scrum developers transparently deliver high-quality, releasable increments of work that meet the needs of the stakeholders? (Choose 1)

- A. Have the team lead personally perform all code merges into the master branch.
- B. Have the software development team strictly adhere to the organization's Definition of Done (DoD).
- C. Reprimand the team when product quality falls below an agreed upon quality gate.
- D. Have the developers send all work through a QA team before release.

## Question 23

The management and development of a key product feature is handled by an external team.

Who represents the progress of this external feature in the Daily Scrum? (Choose 1)

- A. The Product Owner represents external management and development interests.
- B. The Scrum Master represents external management and development interests.
- C. The Team Lead represents external management and development interests.
- D. External development and management issues outside of the control of the Scrum Developers are not presented at the Daily Scrum.

## Question 24

According to the Scrum Guide, which of the following is true about the Increment? (Choose 3)

- A. Each new Increment gets added to all the prior Increments
- B. An Increment doesn't have to be usable.
- C. Only one Increment can be created per Sprint.
- D. An Increment must be compatible with and work with all previous Increments.
- E. An Increment can be seen as a concrete stepping stone toward the Product Goal.

## Question 25

You have just acquired a new company, and two of their Scrum teams will be added to your project to help build the company's flagship product. How should the teams be organized? (Choose 1)

- A. Have the Scrum Master create three new teams with an equal combination of experience, seniority, and technical skill.
- B. Keep the teams in their initial state and allow them to self-organize when the project is complete.
- C. Introduce the teams to each other and have them self-organize into teams of equal size.
- D. Keep the initial structure and layout of the teams, and allow them to self-organize over time.

Be @Scrumtuous and Follow @Xennial on Twitter

## Answers

### Answer 1

The Sprint Planning meeting is one of Scrum's timeboxed events.

What is the timebox for the Sprint Planning meeting?

- A. A minimum of 4 hours
- B. A maximum of 4 hours
- C. A minimum of 8 hours
- D. A maximum of 8 hours

The correct answer is D.

According to the Scrum Guide, "Sprint Planning is timeboxed to a maximum of eight hours for a one-month Sprint. For shorter Sprints, the event is usually shorter."

### Answer 2

The Scrum Guide demands that management allows teams to be self-managed.

What advantages does self-management offer to Scrum Teams and the team's members?

- A. Increased personal accountability with regards to the work developers perform.
- B. Increased accuracy in estimating Product Backlog Item completion dates.
- C. Increased personal commitment to the team and the project.
- D. Increased ingenuity and inventiveness when solving problems.
- E. Increased ability to comply with industry regulation, rules and guidelines.

The correct answers are A, C and D.

Self-management can empower teams by providing them with greater flexibility and autonomy, which can lead to increased motivation, creativity and ownership of work.

Additionally, self-managed teams can better recognize and utilize everyone's strengths, while also facilitating deeper collaboration, shared responsibility and more effective problem solving.

Lastly, self-management can result in faster decision-making as teams have the authority to act in their own best interests.

## Answer 3

What should a Scrum Master do if the Scrum Developers are missing the necessary skills, tools and infrastructure needed to complete the highest-priority backlog items?

- A. Dissolve the Sprint and inform management that the Scrum Team is not ready to use the Scrum framework.
- B. Instruct the Product Owner to accept partially Done Increments for this Sprint.
- C. Change the priority of the current Sprint to improving the development team's skills and acquiring the necessary infrastructure instead of delivering an Increment.
- D. Coach the Scrum Developers on how to spend time during the Sprint learning and improving their skills and acquiring tools they will need to complete important Backlog Items.
- E. Discuss adjusting the Definition of Done to address the tooling issue.

The correct answers are D and E.

The Scrum Developers are expected to have all of the skills required to complete all the Product Backlog items. If those skills do not exist, the developers must acquire them as they work on other Product Backlog items.

It is also possible to adjust the Definition of Done to make the delivery of an Increment possible. For example, if the Definition of Done says every product must be tested by flying it to Mars and back, but you don't have a spaceship that can fly to Mars, it wouldn't be unreasonable to remove that requirement.

Note that adjusting the Definition of Done is not a task to be taken lightly. It should not be adjusted simply to make life easier for the developers or to allow lower-quality work to pass inspection. But if the Definition of Done includes criteria that makes the creation of an inspectable Increment impossible, then it is simply pragmatic to adjust it.

Every Sprint must provide a real, tangible increment of work. There are no 'Infrastructure Sprints', 'Release Sprints' or 'Sprint Zeros' in Scrum. Every Sprint must have the delivery of an increment of work as its goal.

## Answer 4

When should newly identified work or additional information about work be added to the Sprint Backlog?

- A. New insight about the Sprint Plan should be added to the Sprint Backlog immediately.
- B. When the Product Owner agrees that the information should be added to the Sprint Backlog
- C. Sprint Backlog items can only be added during Sprint Planning.
- D. During the Daily Scrum on approval from the team lead

The correct answer is A.

Any time new information that impacts the Sprint Plan is discovered, it should be added to the Sprint Backlog immediately.

Scrum prides itself on *lean thinking*. Waiting to add new details to the Sprint Plan or the Sprint Backlog is not lean. Why wait? It's more lean to do it right now.

## Answer 5

In Scrum, testers, QA personnel, and business analysts on a Scrum Team are considered developers.

- A. True
- B. False

The correct answer is A.

Everyone on a Scrum Team that contributes to building Product Backlog items and helping them to meet the Definition of Done is considered a developer.

According to the Scrum Guide:

"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. Scrum Teams are cross-functional, meaning the members have all the skills necessary to create value each Sprint."

Terms like *Quality Assurance Teams*, *testers*, *team leads*, *business analysts* and *technical writers* never appear at all in the Scrum Guide. If people with these skills are required to complete a PBI and create an Increment that meets the Definition of Done, these people are simply considered developers.

## Answer 6

What effects would be observed on the original Scrum Team when two additional Scrum Teams are included in the development process for the same product?

- A. The original Scrum team's productivity is likely to decrease slightly.
- B. The original Scrum team's productivity is likely to stay the same.
- C. The original Scrum team's productivity is likely to increase slightly.
- D. The original Scrum team's productivity is likely to increase significantly.

The correct answer is A.

The original Scrum Team might experience a period of reduced productivity as the teams become accustomed to working together, but over the medium term, the collective productivity of all the teams should return to a normal baseline.

There are several potential reasons why the productivity of a well-functioning Scrum team might decrease when new teams are added to the project:

### Forming and Storming

When new teams are added to a project, the teams must go through the *forming and storming* stages of group development. This is where they get to know each other and figure out how to work together effectively. This process can take time and can be disruptive to the original team's productivity.

### Communication

Effective communication is essential for multiple Scrum Teams working together to build the same product. When new Teams are added, the original team's communication channels can become disrupted, leading to misunderstandings, delays, and other issues that can impact productivity.

### Disruption to the Processes

Introducing new teams to a product development effort can also disrupt established processes and ways of working, which can lead to confusion and delays.

Adding multiple teams to assist in the development of a product is a significant change, especially if a single team working on the product had established processes and methodologies. Such a change is likely to hurt productivity in the short term, although in the long-term productivity levels should return to normal.

## Answer 7

What should the Scrum Team do with a Product Backlog item that was added to the Sprint Backlog if it does not meet the Definition of Done at the end of a Sprint?

- A. The Product Owner can release it with the permission of the stakeholders.
- B. Return it to the Product Backlog and let the Product Owner reorder it.
- C. Award partial points for the work done so velocity is recorded and return what remains to the Product Backlog.
- D. Do not present the progress made on the incomplete item to the Stakeholders or customers

The correct answers are B and D.

If a Product Backlog item doesn't meet the Definition of Done by the end of the Sprint, which means it's not complete, it goes back into the Product Backlog.

The Scrum Team will then decide if work on the backlog item should continue during the next Sprint, and if so, re-estimate the work needed to complete the undone PBI.

Avoid any answer on the Scrum certification exam that talks about 'points.' There is no mention of 'points' in the Scrum guide.

## Answer 8

During Sprint Planning, Scrum Developers choose the various Product Backlog items they believe they can transform into a usable and valuable Increment before the Sprint ends.

What level of effort and resolve must Scrum developers put into completing the Sprint Backlog items they have chosen for the Sprint?

- A. The developers will work at a sustainable pace in order to complete as many Sprint Backlog items as possible in conformance with the Definition of Done.
- B. The developers must put in all the effort required to successfully complete all Sprint Backlog items before the Sprint is complete.
- C. The developers must commit to delivering core functionality but testing and quality assurance is not required.
- D. The developers must complete the selected Backlog Items in accordance with the Definition of Done, and also provide unit testing, technical documentation and also pass any QA routines.

The correct answer is A.

When it comes to how much effort the team puts into completing Product Backlog item selected for the Sprint, the two key concepts are:

- Working at a sustainable pace.
- Meeting the Definition of Done.

"Working in Sprints at a sustainable pace improves the Scrum Team's focus and consistency."

When a Development Team selects a Product Backlog item for a Sprint, they need to determine how much work they can commit to completing during that Sprint. The amount of work the team commits to is based on their understanding of their own capacity along with their understanding of how much effort is required to meet the agreed upon Definition of Done.

The concept of the Definition of Done is very closely related to the amount of work the developers commit to during a Sprint. The Definition of Done is a shared understanding among the team of the criteria that must be met for a Product Backlog item to be considered complete.

The Definition of Done typically includes criteria related to functionality, quality and usability, among other things. The Increment is complete when the PBI meets the Definition of Done. If unit testing or QA routines are required for a PBI to be complete, those items must be included in the Definition of Done. They are not tasks that are to be completed *after* the Definition of Done has been achieved.

## Answer 9

Who is best equipped to estimate the amount of time required to complete a Product Backlog item(PBI)?

- A. The Product Owner, after adding the PBI to the Product Backlog.
- B. The Scrum Master, after reading the description of the PBI.
- C. The Project Manager, after assigning the PBI to the project.
- D. The individual developer tasked with working on the PBI.
- E. The team of Scrum Developers, after discussions with the Product Owner about the PBI.

The correct answer is A.

Only the developers who do the work can estimate the time required to complete a PBI.

According to the Scrum Guide, "Through discussion with the Product Owner, the Developers select items from the Product Backlog to include in the current Sprint."

## Answer 10

A feedback loop is an opportunity for a team or group to discuss, inspect and most importantly, adapt.

Which of these events provide feedback loops in Scrum?

- A. The Sprint Retrospective
- B. The Daily Standup
- C. The Sprint Review
- D. The Performance Review
- E. The Daily Scrum

The correct answers are A, C and E.

In Scrum, the Sprint Review, Sprint Retrospective, and Daily Scrum are considered feedback loops because they provide opportunities for the Scrum Team to inspect and adapt their work.

The Daily Scrum, which is held every day during the Sprint, is a brief gathering where the developers discuss their progress towards the Sprint Goal. During this meeting, the team members share what they have accomplished since the last meeting, what they plan to accomplish today and any obstacles that may prevent them from achieving their goals. This daily feedback loop allows the team to adjust its plan for the Sprint based on the progress made and any obstacles encountered.

The Sprint Review is the 2nd last event of the Sprint. It is an opportunity for the Scrum Team to review and discuss the work that was completed during the Sprint. Stakeholders get to see completed Product Backlog items that have met the Definition of Done and subsequently provide feedback on that work. The team uses this feedback to adjust for the next Sprint.

The Sprint Retrospective is held at the end of the Sprint, and it is an opportunity for the Scrum Team to reflect on its performance during the Sprint. During this meeting, the team discusses what went well, what did not go well, and how they can improve in the next Sprint. The team uses this feedback loop to make adjustments to its processes which should help improve performance in future Sprints.

## Answer 11

The Scrum Master sees the Product Owner (PO) struggle with the task of ordering the Product Backlog.

What action would a certified Scrum Master take in these circumstances?

- A. Have the Product Owner extend the length of the current Sprint to have more time to order the Product Backlog before the next Sprint begins.
- B. Inform the Product owner that it is the development team's responsibility to order the Product Backlog.
- C. Coach the Product Owner on the importance of ordering the Product Backlog and how the PO is responsible for an order that will deliver the greatest value.
- D. The Scrum Master should provide the Product Owner with the Product Backlog that was ordered by the Scrum Development Team with the Scrum Master's oversight and guidance.
- E. Tell the Product Owner to work with the developers to prioritize Backlog Items based on which items will be the easiest and fastest to implement.

The correct answer is C.

The Scrum Master is a coach and facilitator, and it is the Scrum Master's job to coach and motivate members of the Scrum Team when they need guidance or direction as it pertains to implementing Scrum.

If the Scrum Master sees the Product Owner struggling with the task of ordering the Product Backlog, there are several actions they can take to assist:

### Try and Coach the Product Owner

The Scrum Master can provide coaching to the Product Owner on effective Product Backlog management practices, such as prioritization techniques, clear writing strategies and backlog refinement sessions. They can also guide the PO on how to involve stakeholders in the prioritization process and how to manage dependencies between Product Backlog items.

### Always Encourage Feedback

The Scrum Master can encourage the Product Owner to seek feedback from stakeholders and the development team on the order of Product Backlog items. This feedback can help the Product Owner refine the order and ensure that the team is aligned with the product vision and Product Goal.

Just remember, it's not the Scrum Master's job to do the work of the Product Owner. The order of the product backlog, which is critically important to ensure maximum value gets delivered during

development, is the responsibility of the PO.

The Scrum Master's role is to support the team and help them to achieve their goals. By providing coaching, facilitating collaborative sessions and providing insights and feedback, the Scrum Master can help the Product Owner learn to effectively prioritize and manage the Product Backlog.

## Answer 12

The DevOps team is putting software into production that fails in terms of performance and security. The development team says they only deal with functional requirements, not non-functional requirements like security and performance.

What's the best way for the Scrum Master to deal with this?

- A. Share stories from the help desk about the issues the deployed software is having.
- B. Make sure non-functional requirements are clearly articulated in the Definition of Done.
- C. Run continuous integration tests throughout the Sprint and have developers address performance and security issues that arise before deployment.
- D. Ask the Product Owner to add new Product Backlog items that deal specifically with performance and security.
- E. As the Scrum Master, you should push back on the DevOps team and explain that non-functional requirements are not part of the development team's planning tasks.

The correct answers are B and D.

Developers must address both functional and non-functional requirements if they are part of the product they are building.

To make sure an increment is not released without non-functional requirements being addressed, just add the non-functional requirements to the Definition of Done. That way an increment is not considered complete unless all functional and non-functional requirements have been dealt with.

Furthermore, having the Product Owner add non-functional requirements as backlog items gives them visibility and makes the Product Owner aware of them.

Functional and non-functional requirements are both important aspects of software development that help define what a software system should do and how it should perform. However, there are significant differences between the two types of requirements.

Functional requirements describe what a system should do in terms of specific features, functions, or behaviors. They are typically expressed as specific tasks, actions, or operations that the system should be able to perform. Examples of functional requirements include user authentication, data input and retrieval, reporting, and error handling. Functional requirements are often the most

visible and tangible aspects of a software system and are usually easier to define and test than non-functional requirements.

Non-functional requirements, on the other hand, describe how a system should perform in terms of factors such as reliability, scalability, security, usability, and performance. Non-functional requirements are often more subjective and harder to quantify than functional requirements. They are usually expressed as constraints or quality attributes that the system should exhibit. Examples of non-functional requirements include response time, availability, user experience, and maintainability. Non-functional requirements are important because they help ensure that a system is usable, efficient, and effective, and can be sustained over time.

## Answer 13

Sprints are kept to 30 days or less because:

- A. Risk increases with longer Sprints.
- B. Shorter Sprints are easier to budget.
- C. Complexity is likely to rise with longer Sprints.
- D. Monthly Sprints integrate easier with reporting and tracking tools like JIRA
- E. The likelihood that the Sprint Goal will become invalid increases with longer Sprints.

The correct answers are A, C and E.

Sprints in Scrum are kept to 30 days or less for several reasons related to risk, complexity, and the validity of the Sprint Goal:

Risk: The longer a Sprint runs, the more risk there is that the team will encounter unforeseen issues that may derail the Sprint. By keeping Sprints short, the team can identify and mitigate risks more quickly, reducing the likelihood of a major issue causing a significant delay or complete failure of the Sprint.

Complexity: As the complexity of a Sprint increases, it becomes more difficult to accurately estimate the amount of work that can be completed within a given timeframe. By keeping Sprints short, the team can more accurately estimate the amount of work they can complete, reducing the risk of overcommitting and failing to deliver on the Sprint Goal.

Validity of Sprint Goal: The Sprint Goal is a critical element of any Sprint, and it serves as the guiding focus for the team's work during the Sprint. If the Sprint runs for too long, the Sprint Goal may become less relevant or may even become invalid as the team learns new information or requirements change.

By keeping Sprints short, the team can more effectively focus on achieving the Sprint Goal before it becomes outdated.

## Answer 14

According to the Scrum Guide, the Sprint Backlog is composed of three interdependent parts.

Which three of the following items together make up the Sprint Backlog?

- A. The Sprint Goal
- B. The Product Goal
- C. An actionable plan
- D. Product backlog items chosen for the Sprint
- E. Sprint backlog items not completed on the previous Sprint

The correct answers are A, C and D.

The answer to this question is pulled right out of the Scrum Guide:

"The Sprint Backlog is composed of the Sprint Goal (why), the set of Product Backlog items selected for the Sprint (what), as well as an actionable plan for delivering the Increment (how)."

With regards to the option about 'backlog items not completed on the previous Sprint,' items not completed on a previous Sprint are removed from the Sprint backlog and are returned to the Product Backlog.

They are not automatically added to the next Sprint. You will get tested on that point.

## Answer 15

Who is responsible for removing barriers between stakeholders and Scrum Teams?

- A. The Product Owner
- B. The Scrum Master
- C. The Project Manager
- D. The Development Team

The correct answer is B.

The Scrum Master is responsible for removing barriers between stakeholders and Scrum Teams.

This question is pulled directly out of the Scrum Guide:

"The Scrum Master serves the organization in several ways, including removing barriers between stakeholders and Scrum Teams."

## Answer 16

Sometimes a team has too many developers. Other times a developer just might not be the right fit for a development team.

Who has the power to remove an underperforming developer from a Scrum team?

- A. Only someone in management or Human Resources can remove an underperforming developer from a Scrum team. This is not a Scrum responsibility.
- B. The Scrum Master can remove underperforming developers from a Scrum team.
- C. The responsibility to remove a developer from the Scrum Team lies outside of the scope of the Scrum Guide.
- D. The team of Scrum Developers has the final say over who is allowed to be a member of the development team.

The correct answer is D.

Development teams in Scrum are self-managed and self-organized. If the team decides a member is impeding progress, they have the final say over whether the team member be allowed to stay on the team or not.

Note, this doesn't mean the person is fired. It just means this team was not the correct fit. The person may end up being a great performer somewhere else in the organization.

## Answer 17

Which of the following statements is true about project and Sprint cancellations?

- A. Only the Scrum Master can cancel a Sprint.
- B. Only the Product Owner can cancel a Sprint.
- C. Only the Scrum Master can cancel the project.
- D. Only the Product Owner can cancel the project.

The correct answer is B.

Only the Product Owner can cancel a Sprint.

Neither the Scrum Master nor Product Owner has the authority to cancel a *project*. The Scrum Guide talks about *product* development, not *project* management.

"A Sprint could be canceled if the Sprint Goal becomes obsolete. Only the Product Owner has the authority to cancel the Sprint."

## Answer 18

According to the Scrum Guide, what is the purpose of the Sprint Backlog?

- A. The Sprint Backlog is designed to help user stories get estimated in story points.
- B. The Sprint Backlog is a list of all work the team hopes to complete in the current Sprint.
- C. The Sprint Backlog is a list of all work the team hopes to complete in the next Sprint.
- D. The Sprint Backlog is the Scrum Developers' actionable plan for the Sprint.
- E. The Sprint Backlog is ordered and arranged by the Product Owner to maximize the developer's productivity.

The correct answers are B and D.

According to the Scrum Guide, "The Sprint Backlog is composed of the Sprint Goal (why), the set of Product Backlog items selected for the Sprint (what), as well as an actionable plan for delivering the Increment (how)."

Remember that the Sprint Backlog is all about why, what and how!

Note that all events happen *within* the Sprint. So Sprint Planning happens for the current Sprint, not the 'next' Sprint. Sprint Planning, development, the Sprint Review and the Sprint Retrospective all happen with the boundaries of the same Sprint.

## Answer 19

Which one of the following is the most accurate description of Scrum as defined by the Scrum Guide?

- A. A complete, incremental, and iterative software development process
- B. An incomplete, incremental, and iterative software development process
- C. A purposefully incomplete, lightweight framework to help find solutions for complex problems
- D. A purposefully complete, lightweight framework to help find solutions for complex problems

The correct answer is C.

Scrum is not just for the domain of software development, so any definitions that try to pin Scrum to the software development world are incorrect.

Scrum self-describes as being a purposefully incomplete, lightweight framework designed to help find solutions for complex problems.

From the Scrum Guide: "Scrum is a lightweight framework that helps people, teams, and organizations generate value through adaptive solutions for complex problems. The Scrum framework is purposefully incomplete, only defining the parts required to implement Scrum theory."

## Answer 20

Three Scrum Teams are working on the same product. Each team produces an increment of work by the end of their Sprint. When should the Increment from a given team be integrated into the project?

- A. Teams should be doing continuous integration, and increments must be integrated with the work of other teams before the Sprint review.
- B. Teams should be doing continuous integration, and increments must be integrated with the work of other teams after the Sprint review.
- C. Three Scrum teams working on the same project should be independent of each other and not integrate their work.
- D. The three teams should only integrate their code at the end of the project.

The correct answer is A.

The increments from multiple Scrum Teams working on the same product should be integrated continuously. If they are not integrated continuously, at the very least, they need to be integrated before the Sprint Review.

If code changes are not integrated, how does anyone know that the changes and updates they made even work?

Multiple Scrum teams working on the same project should continuously integrate their increments of work for several reasons:

- Early detection of integration issues
- Improved transparency
- Teams will collaborate and communicate more often

If something works and meets the Definition of Done, it should be integrated right away. Waiting makes no sense. Waiting is certainly not in line with Scrum's commitment to lean thinking.

## Answer 21

Who holds developers accountable for creating a plan, instilling quality, and adhering to the Definition of Done?

- A. The Product Owner holds the developers accountable
- B. The Scrum Master holds the developers accountable
- C. The Stakeholders hold the developers accountable
- D. Developers hold each other accountable as professionals

The correct answer is D.

This answer to this question comes right out of the Scrum Guide:

"The specific skills needed by the Developers are often broad and will vary with the domain of work. However, the Developers are always accountable for:

- Creating a plan for the Sprint, the Sprint Backlog;
- Instilling quality by adhering to a Definition of Done;
- Adapting their plan each day toward the Sprint Goal; and,
- **Holding each other accountable as professionals.** "

## Answer 22

What is the best way for the Product Owner to ensure the Scrum developers transparently deliver high-quality, releasable increments of work that meet the needs of the stakeholders?

- A. Have the team lead personally perform all code merges into the master branch.
- B. Have the software development team strictly adhere to the organization's Definition of Done (DoD).
- C. Reprimand the team when product quality falls below an agreed upon quality gate.
- D. Have the developers send all work through a QA team before release.

The correct answer is B.

The Product Owner needs everyone to adhere to the Definition of Done because it helps ensure that the team delivers high-quality Increments of work that meet the needs of the stakeholders.

The Definition of Done is a shared understanding between the Product Owner, stakeholders and the developers about what it means for a Product Backlog item to be considered complete. It includes a set of criteria that must be met for each item such as passing tests, meeting performance

standards and conforming to coding standards.

By adhering to the Definition of Done, the development team ensures that they are producing work that is of high quality and meets the expectations of the Product Owner. This, in turn, helps the Product Owner manage stakeholder expectations as they can rely on the Scrum Developers to consistently deliver work that meets the agreed-upon criteria.

Moreover, adhering to the Definition of Done promotes transparency and accountability. The developers commit to delivering work that meets the agreed-upon criteria, which helps to ensure that they take ownership of the quality of their work. The Product Owner can also track progress more accurately as they can trust that items that meet the Definition of Done are truly complete.

## Answer 23

The management and development of a key product feature is handled by an external team.

Who represents the progress of this external feature in the Daily Scrum?

- A. The Product Owner represents external management and development interests.
- B. The Scrum Master represents external management and development interests.
- C. The Team Lead represents external management and development interests.
- D. External development and management issues outside of the control of the Scrum Developers are not presented at the Daily Scrum.

The correct answer is D.

Work produced by external teams is not a topic Scrum Developers should focus on during the Daily Scrum. The Scrum Developers should only be focused on their own product-related activities, the Sprint Goal and how they can adapt their current plan to meet the Sprint Goal.

*Focus* is one of the five Scrum values.

Something created and developed outside the Scrum Team should not be part of their backlog as they have no control over it.

The Scrum Team is not subject to external management but is instead empowered to self-manage.

From the Scrum Guide: "The Scrum Team is responsible for all product-related activities from stakeholder collaboration, verification, maintenance, operation, experimentation, research and development, and anything else that might be required. They are structured and empowered by the organization to manage their own work."

## Answer 24

According to the Scrum Guide, which of the following is true about the Increment?

- A. Each new Increment gets added to all the prior Increments
- B. An Increment doesn't have to be usable.
- C. Only one Increment can be created per Sprint.
- D. An Increment must be compatible with and work with all previous Increments.
- E. An Increment can be seen as a concrete stepping stone toward the Product Goal.

The correct answers are A, D and E.

An Increment must be usable and multiple Increments can be created per Sprint, which makes those two options wrong.

The correct options come directly out of the Scrum Guide's definition of a Sprint:

"An Increment is a concrete stepping stone toward the Product Goal. Each Increment is additive to all prior Increments and thoroughly verified, ensuring that all Increments work together. To provide value, the Increment must be usable."

## Answer 25

You have just acquired a new company, and two of their Scrum teams will be added to your project to help build the company's flagship product. How should the teams be organized?

- A. Have the Scrum Master create three new teams with an equal combination of experience, seniority, and technical skill.
- B. Keep the teams in their initial state and allow them to self-organize when the project is complete.
- C. Introduce the teams to each other and have them self-organize into teams of equal size.
- D. Keep the initial structure and layout of the teams, and allow them to self-organize over time.

The correct answer is D.

Breaking up teams and reassembling them can be disruptive, as it may require additional time for the new teams to get up to speed and establish a working relationship. This disruption can result in delays in project delivery, increased project costs and decreased quality of work.

There's no need to break the teams up right away. Let them get to know each other and allow them to self-organize over time.

Be @Scrumtuous and Follow @Xennial on Twitter

## Final Exam: 40 Questions



**Good Luck!**

## Question 1

Halfway through the Sprint the development team has become uncomfortable with the Sprint Goal and feels a change needs to be made to bring it in line with the Sprint Plan.

According to the Scrum Guide, when can the Sprint Goal be changed or adjusted? (Choose 1)

- A. The Sprint Goal can be changed or adjusted any time the Product Goal is changed or adjusted.
- B. The Sprint Goal can be changed or adjusted any time during the Daily Scrum.
- C. A Sprint Goal can be changed or adjusted any time during the Sprint.
- D. The Sprint Goal must be finalized before the end of Sprint Planning and it cannot be changed throughout the Sprint.
- E. The Sprint Goal must be finalized before the development Sprint starts and it cannot be changed throughout the Sprint.

## Question 2

How might a Scrum Master help the Scrum Team resolve an internal disagreement that has caused inter-personal conflicts between team members. (Choose 3)

- A. Use various coaching and facilitation techniques such as open feedback and active listening.
- B. Involve everyone on the team of Scrum Developers in forming a resolution that is in line with the Scrum values.
- C. Ask an external consultant on the production team to determine the most sensible path forward.
- D. Incorporate the Human Resources (HR) department to help resolve any inter-personal conflicts that have arisen.
- E. Coach the team on maintaining a psychologically safe environment for all participants.

## Question 3

If the company you work for does not have an organizational standard that defines the Definition of Done, whose responsibility is it to provide a definition before development begins? (Choose 1)

- A. The development team
- B. The Product Owner
- C. The Scrum Team, which includes the Scrum Master and Product Owner
- D. The Definition of Done is formed from a consensus between the Scrum Master and the Product Owner

## Question 4

Under which circumstance can the Scrum Master extend the length of the Sprint once the Sprint has started? (Choose 1)

- A. The Sprint can be extended equal to the number of days a Product Owner is absent
- B. The Sprint can be extended equal to the number of holidays that occurred during the Sprint
- C. The Sprint length can be changed any time prior to the end of the Sprint Planning meeting
- D. The Sprint length cannot be changed after a Sprint has started

## Question 5

The development team failed to complete all the items in the Sprint Backlog before the end of the Sprint. What happens next? (Choose 1)

- A. The Scrum Master extends future Sprints by the amount of time it would have required to complete all Sprint Backlog items.
- B. The length of the Sprint remains the same, selected Sprint Backlog items get returned to the Product Backlog, and the Scrum Team tries to learn and adapt from the experience.
- C. The current Sprint is temporarily extended to allow all Sprint Backlog items to be completed. Insight is garnered at the Sprint Retrospective and the team adapts to avoid this situation in the future.
- D. The Sprint Review is postponed until all selected Sprint Backlog items are completed.

## Question 6

What does the slope of the line in a burndown chart demonstrate? (Choose 1)

- A. The percentage of the budget consumed by the project over time.
- B. The number of Story Points needed per Sprint to complete the project on time.
- C. Changes in the team's velocity.
- D. The amount of work that has been completed compared to how much more work needs to get done.

## Question 7

Two teams working on the same Product should maintain separate Product Backlogs. (Choose 1)

- A. True
- B. False

## Question 8

You, the Scrum Master, have been told that the quality assurance (QA) tests performed by an external team have been delayed, and due to this delay, it's unlikely the Definition of Done will be achieved before the end of this 3-week Sprint. However, if the Sprint was extended a week, the QA work would likely be done.

How would you proceed as a Scrum Master? (Choose 2) (Choose 2)

- A. Extend the Sprint a week to allow the QA team to complete their work.
- B. Investigate why the Scrum Team does not have all the cross-functional skills to achieve the Definition of Done without an external QA team
- C. Remove QA work from the Definition of Done for this Sprint.
- D. Explain to the organization that Scrum is not designed to manage the work performed by people external to the Scrum Team.
- E. Turn the next Sprint into a 'Performance and Security' sprint and focus on these two non-functional aspects of the code.

## Question 9

Self-organization and self-management are two important attributes of a Scrum Team.

What is most important criteria for developers to think about when they self-organize into development teams? (Choose 1)

- A. Does each team have a strong and competent team lead?
- B. Is there a good balance of senior and junior developers?
- C. Will each team be able to form a sub-group for QA and testing?
- D. Will the team have the cross-functional skills needed to build the product?

## Question 10

Scrum practitioners are encouraged to focus on the current Sprint, and while the Product Goal should always be 'top of mind', Scrum does not plan two or three Sprints into the future. This is an example of which Scrum value? (Choose 1)

- A. Courage.
- B. Respect.
- C. Focus.
- D. Openness.
- E. Honesty.

## Question 11

According to the Scrum Guide, which of the following are duties of a Scrum Master? (Choose 1)

- A. Participating as an equal member of the Scrum team while also acting as a leader who serves.
- B. Assigning action items for team members to complete during the Sprint.
- C. Communicating the development team's answers to stakeholder questions through transcription, email or direct communication.
- D. Taking dictation during the Daily Scrum and reporting discussions that take place during the Daily Scrum back to management.
- E. Scheduling meetings and booking conference rooms for the team.

## Question 12

Which of the following Scrum events are timeboxed to limit the amount of time spent on them? (Choose 3)

- A. Standup Meetings.
- B. The Release Sprint.
- C. The Sprint Retrospective.
- D. Sprint Planning.
- E. Stakeholder Meetings.
- F. Sprint Zero.
- G. The Daily Scrum.

## Question 13

Which of the following are key responsibilities of the Scrum Master on a Scrum Team? (Choose 3)

- A. The Scrum Master helps employees and stakeholders understand and enact an empirical and lean approach to solving complex problems.
- B. The Scrum Master assigns Product Backlog items to developers when they are idle.
- C. The Scrum Master helps those outside the Scrum team understand Scrum and how to interact with teams that employ the Scrum framework.
- D. The Scrum Master ensures that all Scrum events are kept within their timebox.
- E. The Scrum Master demos to stakeholders the features that have met the Definition of Done during the Sprint Review.

## Question 14

For an Increment of work to be demonstrated to the stakeholders at the Sprint Review, it must be:  
(Choose 3)

- A. Valuable
- B. Paid for
- C. Peer reviewed
- D. Usable
- E. Compatible with past Increments

## Question 15

The Scrum Guide very briefly mentions burn-down and burn-up charts.

What does a trend line displayed on a burndown chart indicate about the team's progress? (Choose 1)

- A. How quickly the team is 'burning through' money budgeted for the project.
- B. How quickly other Scrum teams on the same project are 'burning through' Product Backlog items compared to your team.
- C. The amount of work remaining plotted against time, which predicts when the project will finish if nothing changes with the Product Owner, Product Backlog items or the development team.
- D. How quickly the project will be completed if Product Backlog items are slowly removed by the Product Owner.

## Question 16

When should new developers be added to a Scrum team? (Choose 1)

- A. During Sprint Zero the team size should be set for the duration of the project.
- B. After the current Sprint ends but before the next Sprint begins.
- C. When adding a new developer will help development continue at a sustainable pace.
- D. When the Project Manager provides enough budget money for a new developer.

## Question 17

The project has kicked off and the first Sprint was a great success.

According to the Scrum Guide, when does the second Sprint begin? (Choose 1)

- A. Immediately after the Sprint Review for the first Sprint is completed.
- B. Immediately after Sprint Planning for the second Sprint is completed.
- C. The second Sprint begins immediately after the first Sprint is finished.
- D. When the Scrum Master declares the start of the new Sprint in JIRA.

## Question 18

The development team can't agree whether to use Java or Python to develop microservices for their current project.

How can the Scrum Master help the Scrum Team resolve an internal disagreement about whether to build in Java or Python? (Choose 2)

- A. Have the developers consult an external, impartial expert on the topic and agree to go with the external expert's decision.
- B. Have the development team argue both sides to the Scrum Master and have the Scrum Master come to a final, impartial decision.
- C. Use coaching techniques like reflective listening and visualization to help guide the entire development team towards building a consensus.
- D. Teach the development team about collaborative techniques to build consensus, such as actively listening to others and paraphrasing contrary positions.

## Question 19

When is a Product Backlog item (PBI) considered fully complete and ready for release? (Choose 1)

- A. A Product Backlog item is complete when a Quality Assurance team verifies that the work passes all acceptance criteria, making it ready for release
- B. A Product Backlog item is complete when all work in the Sprint Backlog related to the item is finished, making it ready for release
- C. A Product Backlog item is complete when it meets the Definition of Done and further Increments of work on it are no longer possible, making it ready for release.
- D. Product Backlog items are never officially complete until the Product Goal is achieved.

## Question 20

Who on the Scrum Team has the responsibility of ordering the Product Backlog? (Choose 1)

- A. The Product Owner orders the Product Backlog, with the items that deliver the most value to the stakeholders at the top.
- B. The Development Team orders the Product Backlog, with items listed chronologically according to what the development team should logically build first.
- C. The Scrum Master orders the Product Backlog, with the items that the Product Owner says will deliver the most value to the stakeholders at the top.
- D. The Product Owner prioritizes the Product Backlog, with PBIs with the highest story-points assigned to them first.

## Question 21

What are the two primary responsibilities of a new Product Owner when taking ownership of a product that is currently under development with a long history of multiple, successful Sprints? (Choose 2)

- A. Ensuring that the development teams know which Product Backlog items provide the greatest value and should be developed first
- B. Describing and fully decomposing product features in Use Case and story form.
- C. Providing detailed development specs and guidelines to the development team.
- D. Interacting with stakeholders to learn how to represent their needs in the Product Backlog.
- E. Creating both functional and non-functional tests to validate the Increment meets the Definition of Done.

## Question 22

Under what circumstances might the decision be made to prematurely terminate a Sprint? (Choose 1)

- A. When a self-managed team of Scrum Developers realizes the work committed to during Sprint planning cannot possibly be delivered by the end of the Sprint.
- B. When the Sprint Goal no longer makes sense and becomes obsolete.
- C. When the Scrum Development team's work needs to be redirected to new opportunities.
- D. When the Product Owner reprioritized the Product Backlog and PBIs currently under development are no longer priorities.

## Question 23

Corporate stakeholders are very busy, as multiple projects are going online this quarter.

The stakeholders have asked to attend every-other Sprint Review to minimize the number of meetings on their calendars. How do you as a Scrum Master respond? (Choose 2)

- A. Coach the stakeholders and the team on how to make the meetings more productive and positive.
- B. Agree with the request of the stakeholders, as reducing meetings is in line with Agile principles.
- C. Educate stakeholders on the importance of Sprint Reviews and work to facilitate more engaging Sprint Reviews.
- D. Consult privately with the Product Owner and see if they will agree to fewer meetings.

## Question 24

As the Scrum Master, it is clear to you that the Sprint Goal is not going to be achieved if the Sprint Backlog isn't adjusted and new Product Backlog items aren't added.

In Scrum, who has the right to add new Product Backlog items to the Sprint Backlog? (Choose 1)

- A. The Product Owner prioritizes Product Backlog items and has the authority to add PBIs to the Sprint Backlog.
- B. The development team owns the Sprint Backlog and can add items to it at any time during the Sprint.
- C. Both the Scrum Master and the developers can add to the Sprint Backlog if changes need to be made to keep the Sprint Goal in focus.
- D. Anyone on the Scrum Team can add or remove items from the Sprint Backlog.

## Question 25

The Product Owner often has no idea how complicated a Product Backlog item will be to build.

Whose responsibility is it to decompose long, epic, Product Backlog items into easily digestible work items that should take no more than a day to complete? (Choose 1)

- A. The Business Analysts.
- B. The Product Owner.
- C. The Scrum Master.
- D. The Team Lead.
- E. The Developers on the team.

## Question 26

Scrum doesn't work if there isn't buy-in from the entire organization, including management.

How do members of the management group in an organization that sponsors a development project support a Scrum Team? (Choose 3)

- A. The organization empowers the Scrum Team to manage their own work.
- B. The organization responds quickly to daily status updates from the Scrum Master.
- C. The organization must provide Scrum Teams with resources that allow for continual improvement.
- D. The organization fully respects and trusts the Product Owner's decisions.

## Question 27

The Scrum team needs to hire an outside specialist, external to the Scrum team, to deliver wireframes needed in order to complete a Product Backlog item and meet the Definition of Done. Some of the developers have complained to you, the Scrum Master, in confidence about the outside specialist.

What are your main concerns in this situation? (Choose 2)

- A. You are concerned that the external dependency may impact software development velocity.
- B. You are concerned that team members are coming to you with complaints rather than figuring out a solution themselves within the team.
- C. You are concerned about the ability of the development team to produce a fully integrated Increment when meeting the Definition of Done depends on work produced by someone outside of the Scrum Team.
- D. You are worried that there will be software developers sitting idle while work is outsourced.

## Question 28

Which of the following is *not* true of the Daily Scrum? (Choose 1)

- A. The Daily Scrum is used to identify impediments.
- B. The Daily Scrum eliminates the need for other meetings.
- C. The Daily Scrum is timeboxed to 30 minutes.
- D. The Daily Scrum encourages quick decision-making.

## Question 29

What are the skills and capabilities the developers on the Scrum Team should have in order to ensure effective performance throughout the Sprint?

(Choose 1)

- A. The ability to decompose Product Backlog items (PBIs) and progressively create Increments of work until a functional release is possible.
- B. The ability to perform all of the core software development work, with the exception of any specialized testing that would require additional tools and staging environments.
- C. The ability to complete a development project within budget and within the timeline agreed upon with the Product Owner.
- D. The ability to take direction from the Scrum Master and complete Product Backlog items assigned to them by the Product Owner.

## Question 30

The development team bit off more than it could chew during Sprint Planning.

They've just told you, the Scrum Master, that there's no way they are even going to come close to completing the Sprint Backlog.

What should you, as the Scrum Master, do? (Choose 1)

- A. Change the Definition of Done and negotiate the quality standard with the Product Owner to ensure all Sprint Backlog items are completed before the end of the Sprint.
- B. While the Sprint is ongoing, have the Product Owner and the developers negotiate the scope of the work being done so as not to endanger the Sprint Goal.
- C. Scale out and assign some of the Sprint Backlog items to an external Scrum Team.
- D. Inform the Product Owner prior to the Sprint Review so the stakeholders won't be disappointed.

## Question 31

What should be done with a Product Backlog item (PBI) that did not meet the Scrum Team's agreed upon Definition of Done by the end of a Sprint? (Choose 2)

- A. Return the item to the Product Backlog.
- B. With the consent of the Product Owner and stakeholders, the PBI can be approved and released.
- C. Automatically add the unfinished PBI to the next Sprint's backlog.
- D. Make sure the team knows that work on this PBI will not be included in the current Sprint's Increment.

## Question 32

Teams sometimes struggle to come up with items to discuss during the Sprint Retrospective.

Which of the following topics and issues would be worthwhile for a Scrum Team to discuss during a Sprint Retrospective?

(Choose 1)

- A. Whether the Definition of Done needs to be updated.
- B. Whether the length of the Sprint needs adjusting.
- C. How to better decompose Product Backlog items during Sprint Planning.
- D. How to improve communication between on-site and remote workers.
- E. All of the above.

## Question 33

Which of the following actions should the Scrum Master never do? (Choose 4)

- A. Tell the Scrum Developers how to manage their workloads.
- B. Extend the length of the Sprint.
- C. End the Sprint early.
- D. Coach the Development Team to ensure the Daily Scrum is timeboxed to 15 minutes.
- E. Reprimand developers who underperform.

## Question 34

Sprint Planning is the first of the four timeboxed events to happen during the Sprint.

According to the Scrum Guide, which topics are to be addressed by the Scrum Team during Sprint Planning? (Choose 3)

- A. How on Earth is the work chosen for the Sprint going to get done?
- B. Why is this a valuable Sprint?
- C. What do you think can get done before the end of this Sprint?
- D. Why were certain Sprint Backlog items not completed in the prior Sprint?
- E. When will infrastructure required for deployment will be ready?

## Question 35

According to the Scrum Guide, which two of the following statements are true about the Product Owner? (Choose 2)

- A. The Product Owner may delegate the creation of Product Backlog items to an associate.
- B. The Product Owner may be a committee or collection of individuals who collaboratively perform the role.
- C. The organization must fully respect and trust the independent decisions of the Product Owner.
- D. The Product Owner should coach the developers in self-management and cross-functionality.
- E. The Product Owner must attend and participate in the Daily Scrum.

## Question 36

The team's developers are amazing! Maybe too amazing?

It's not even half-way through the Sprint and the developers are telling you, the Scrum Master, that they have not selected enough items for the Sprint. Everything is practically done, which will result in developers being underutilized with nothing to do for the rest of the Sprint.

What should the Scrum Master do? (Choose 1)

- A. End the Sprint and reconvene a new Sprint Planning meeting.
- B. Have the Product Owner add new Product Backlog items to the Sprint Backlog.
- C. Create a more rigorous Definition of Done so that quality will improve and it will require more work to complete the items currently in the Sprint Backlog
- D. Have the developers adapt their Sprint Plan, preferably with the help of the Product Owner, and add additional, high-value Product Backlog items to the Sprint Backlog.

## Question 37

During every Sprint, the team attempts to create at least one Increment to present to stakeholders during the Sprint Review.

Why might a Product Owner refuse to immediately release an Increment into production after a successful Sprint Review? (Choose 1)

- A. Increments are automatically released into production so it is not possible to refuse a release.
- B. The Product Owner needs confirmation from the Scrum Master in order to release an Increment into production.
- C. The immediate timing of the release does not make sense for customers or stakeholders.
- D. The Product Owner never refuses to release an Increment into production.

## Question 38

The developers are new to Scrum and they've asked you, an experienced Scrum Master, who should start, lead and manage the Daily Scrum. How do you respond? (Choose 2)

- A. Explain that the most senior developer on the team should start and lead the Daily Scrum
- B. Explain that the most senior developer on the team leads the Daily Scrum while the most junior developer speaks first
- C. Explain that the team lead leads the Daily Scrum while the most senior developer speaks first
- D. Explain how the developers can choose whatever structure they want for the Daily Scrum, so who starts it is up to them.
- E. Explain that Scrum is hierarchy-less and it doesn't require any specific individual to 'lead' during the Daily Scrum.

## Question 39

You are building a limo for a head of state, and security and protection of the occupant are two important, non-functional requirements that must be prioritized at every step in the development process.

What is the best way to ensure security and protection of the occupant are prioritized? (Choose 2)

- A. Outsource security and protection concerns to an external third party that specializes in these areas.
- B. Add a Sprint prior to the release of the car that deals exclusively with security and protection.
- C. Build a special sub-team on the development team that deals exclusively with security and protection of the occupant.
- D. Have the Product Owner add the features that pertain to security and protection of the occupant to the Product Backlog.
- E. Add concerns related to the security and protection of the occupant to the Definition of Done.

## Question 40

Which of the following is true about the PO's attendance at the Daily Scrum? (Choose 1)

- A. The Product Owner must always attend the Daily Scrum to help remove any impediments that exist that jeopardize the Sprint Goal
- B. The Product Owner must always attend the Daily Scrum so they can quickly answer any questions the Scrum Developers have about items they are working on.
- C. The Product Owner must always attend the Daily Scrum to present the stakeholder's point of view.

- D. The Product Owner doesn't need to attend the Daily Scrum.

## Answers

### Answer 1

Halfway through the Sprint the development team has become uncomfortable with the Sprint Goal and feels a change needs to be made to bring it in line with the Sprint Plan.

According to the Scrum Guide, when can the Sprint Goal be changed or adjusted?

- A. The Sprint Goal can be changed or adjusted any time the Product Goal is changed or adjusted.
- B. The Sprint Goal can be changed or adjusted any time during the Daily Scrum.
- C. A Sprint Goal can be changed or adjusted any time during the Sprint.
- D. The Sprint Goal must be finalized before the end of Sprint Planning and it cannot be changed throughout the Sprint.
- E. The Sprint Goal must be finalized before the development Sprint starts and it cannot be changed throughout the Sprint.

The correct answer is D.

The Sprint Goal must be finalized before the end of Sprint Planning and it cannot be changed throughout the Sprint.

According to the Scrum Guide, during Sprint planning, "the whole Scrum Team collaborates to define a Sprint Goal that communicates why the Sprint is valuable to stakeholders. The Sprint Goal must be finalized prior to the end of Sprint Planning."

Note that there are no 'development Sprints' in Scrum. There are just Sprints. There are also no:

- Sprint Zeros
- Release Sprints
- Infrastructure Sprints
- Integration Sprints

Furthermore, all of the Scrum Events take place within a Sprint.

- The Sprint does not start after Sprint Planning.
- The Sprint Review does not happen after the Sprint finishes

A Sprint starts, and then all of the events take place within that Sprint. And when a Sprint ends after the Sprint Retrospective concludes, a new Sprint begins immediately. There is no time between Sprints in Scrum.

## Answer 2

How might a Scrum Master help the Scrum Team resolve an internal disagreement that has caused inter-personal conflicts between team members.

- A. Use various coaching and facilitation techniques such as open feedback and active listening.
- B. Involve everyone on the team of Scrum Developers in forming a resolution that is in line with the Scrum values.
- C. Ask an external consultant on the production team to determine the most sensible path forward.
- D. Incorporate the Human Resources (HR) department to help resolve any inter-personal conflicts that have arisen.
- E. Coach the team on maintaining a psychologically safe environment for all participants.

The correct answers are A, B and E.

A Scrum Team is expected to be cross-functional and have all the skills needed to address issues that arise. The team should be able to make independent decisions without consulting external specialists.

A self-managed, cross-functional team is also expected to resolve conflicts on their own. From the Scrum perspective, and certainly on the Scrum Master certification exam, going to Human Resources to resolve a problem is not a prescribed approach.

The Scrum Guide never once mentions the term "Human Resources," nor is it even reasonable to assume that every team doing Scrum would have a Human Resources department. Avoid any options on a Scrum Master exam question that references HR.

A Scrum Master *coaches and facilitates* in order to help the development team come to their own conclusions and find paths to help them move forward.

## Answer 3

If the company you work for does not have an organizational standard that defines the DoD, whose responsibility is it to provide a definition before development begins?

- A. The development team
- B. The Product Owner
- C. The Scrum Team, which includes the Scrum Master and Product Owner
- D. The DoD is formed from a consensus between the Scrum Master and the Product Owner

The correct answer is C.

According to the Scrum Guide:

"If the Definition of Done for an increment is part of the standards of the organization, all Scrum Teams must follow it as a minimum.

If it is not an organizational standard, the Scrum Team must create a Definition of Done appropriate for the product."

In Scrum, the Definition of Done (DoD) is a shared understanding of what it means for a Product Backlog item to be considered complete and ready to be integrated into the product. The Scrum Team, which includes the Product Owner, the development team and the Scrum Master, collaborates to create the Definition of Done if an organizational standard for the DoD does not already exist.

Almost everything that happens in Scrum happens within the confines of a Sprint. The Definition of Done is one of the few items references in the Scrum Guide that should exist before the first Sprint begins. After all, a team can't really do any Sprint Planning without an established Definition of Done.

After development starts, discussions about updating or adjusting the Definition of Done should happen during the Sprint Retrospective.

Ultimately, the Definition of Done should be agreed upon by the entire Scrum Team and should reflect the standards and expectations for delivering a high-quality product to the customer.

## Answer 4

Under which circumstance can the Scrum Master extend the length of the Sprint once the Sprint has started?

- A. The Sprint can be extended equal to the number of days a Product Owner is absent
- B. The Sprint can be extended equal to the number of holidays that occurred during the Sprint
- C. The Sprint length can be changed any time prior to the end of the Sprint Planning meeting
- D. The Sprint length cannot be changed after a Sprint has started

The correct answer is D.

Once a Sprint has started, the Sprint length cannot be changed.

Changes to the length of the Sprint should be agreed upon by a consensus of the team.

The best time to discuss changing the length of the Sprint is during the Sprint Retrospective. Any changes made to the Sprint length during a Sprint would apply only to future Sprints, not the current Sprint. Once a Sprint starts, its length cannot be changed.

## Answer 5

The development team failed to complete all the items in the Sprint Backlog before the end of the Sprint. What happens next?

- A. The Scrum Master extends future Sprints by the amount of time it would have required to complete all Sprint Backlog items.
- B. The length of the Sprint remains the same, selected Sprint Backlog items get returned to the Product Backlog, and the Scrum Team tries to learn and adapt from the experience.
- C. The current Sprint is temporarily extended to allow all Sprint Backlog items to be completed. Insight is garnered at the Sprint Retrospective and the team adapts to avoid this situation in the future.
- D. The Sprint Review is postponed until all selected Sprint Backlog items are completed.

The correct answer is B.

There is no changing the Sprint length during an active Sprint.

During the Sprint Retrospective, a Scrum Team can discuss the Sprint length and possibly adjust the length of future Sprints, but the length of an active Sprint is fixed and cannot be changed. Sprint Reviews and Sprint Retrospectives will occur at their scheduled time and place.

It's not uncommon for some Sprint Backlog items to go uncompleted during a Sprint. When that happens, they just go back into the Product Backlog.

The team can always discuss why their estimates were off, or why they couldn't complete as many PBIs as they wanted to during the Sprint Retrospective. The Scrum Team should turn the situation into a learning experience and adapt accordingly. That's what Scrum is all about!

## Answer 6

What does the slope of the line in a burndown chart demonstrate?

- A. The percentage of the budget consumed by the project over time.
- B. The number of Story Points needed per Sprint to complete the project on time.
- C. Changes in the team's velocity.
- D. The amount of work that has been completed compared to how much more work needs to get done.

The correct answer is D.

A trend line of a release burndown chart indicates how fast work is being completed relative to the original plan, and can help visualize progress toward completion of the component being graphed, be it a Sprint or a decomposed Product Backlog item.

"A burn down chart enables the team to visualize the progress on a project. A burn down chart includes work remaining on the Y axis against a timeline of the project on the X axis. Depending on the project, these charts can measure work as user stories, tasks, backlogs or other metrics."

-Project Management Tools, TechTarget, Jan 2020. M. Courtemanche.

The Scrum Guide only makes a short reference to burndown charts, and even then, it says they are not a replacement for empiricism. You don't need to be a burndown chart expert, but knowing what one is will keep you from being caught off guard on the Scrum certification exam.

## Answer 7

Two teams working on the same Product should maintain separate Product Backlogs.

- A. True
- B. False

The correct answer is B.

This is false.

Two teams working on the same product will work off the same Product Backlog. They will also share the same Product Goal, Definition of Done and Product Owner.

## Answer 8

You, the Scrum Master, have been told that the quality assurance (QA) tests performed by an external team have been delayed, and due to this delay, it's unlikely the Definition of Done will be achieved before the end of this 3-week Sprint. However, if the Sprint was extended a week, the QA work would likely be done.

How would you proceed as a Scrum Master? (Choose 2)

- A. Extend the Sprint a week to allow the QA team to complete their work.
- B. Investigate why the Scrum Team does not have all the cross-functional skills to achieve the Definition of Done without an external QA team
- C. Remove QA work from the Definition of Done for this Sprint.
- D. Explain to the organization that Scrum is not designed to manage the work performed

by people external to the Scrum Team.

- E. Turn the next Sprint into a 'Performance and Security' sprint and focus on these two non-functional aspects of the code.

The correct answers are B and D.

A Scrum Team must have all of the skills necessary to complete the Definition of Done. The Scrum Team cannot outsource work that is part of their Definition of Done and still have that work managed through Scrum.

Extending the Sprint is never an option. The Sprint length is fixed when the Sprint starts.

The Definition of Done can always be discussed and updated. However, the Scrum Master does not have the right to unilaterally change the Definition of Done, and changes to the Definition of Done are best discussed in the Sprint Retrospective. Furthermore, the Definition of Done should not be changed midway through a Sprint to accommodate moving goalposts. Doing so would be antithetical to Scrum.

Of course, there is nothing in the Scrum Guide that explicitly forbids updating the Definition of Done mid-Sprint, but just keep in mind that all estimates and planning were made given the DoD that existed during Sprint Planning. A mid-Sprint change to the DoD has the potential to throw off all of those estimates and put the Sprint Goal at risk.

Changes to the Definition of Done are best to take place during the Sprint Retrospective.

## Answer 9

Self-organization and self-management are two important attributes of a Scrum Team.

What is most important criteria for developers to think about when they self-organize into development teams?

- A. Does each team have a strong and competent team lead?
- B. Is there a good balance of senior and junior developers?
- C. Will each team be able to form a sub-group for QA and testing?
- D. Will the team have the cross-functional skills needed to build the product?

The correct answer is D.

There are no subteams, titles or hierarchies on Scrum Teams.

All that matters when a group of Scrum Developers is assembled is whether they have, or do they have the ability to acquire, the cross-functional skills matrix needed to build the product under development and achieve the Product Goal.

From the Scrum Guide: "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. Scrum Teams are cross-functional, meaning the members have all the skills necessary to create value each Sprint. They are also self-managing, meaning they internally decide who does what, when, and how."

## Answer 10

Scrum practitioners are encouraged to focus on the current Sprint, and while the Product Goal should always be 'top of mind', Scrum does not plan two or three Sprints into the future. This is an example of which Scrum value?

- A. Courage.
- B. Respect.
- C. Focus.
- D. Openness.
- D. Honesty.

The correct answer is C.

One of the key principles of Scrum is to maintain a clear focus on the current Sprint and avoid distractions from other work that is not part of the Sprint. This principle of focusing on the Sprint is intended to ensure that the team can deliver high-quality work and achieve its Sprint Goal without being sidetracked by other priorities.

This is also a recognition of the fact that things can change quickly, especially after a Sprint Review or Sprint Retrospective, so planning beyond the current Sprint is often a complete waste of time.

By not looking beyond the current Sprint, the development team is able to maintain focus and avoid getting distracted by future work that may not be relevant or may change over time. This helps the team to remain agile and respond quickly to changes in the market or in customer needs, since they are not locked into a fixed plan that may become obsolete.

Moreover, the Scrum framework provides regular opportunities to inspect and adapt the work done in the current Sprint and adjust the backlog accordingly. This allows the team to constantly improve the product and respond to feedback from stakeholders.

## Answer 11

According to the Scrum Guide, which of the following are duties of a Scrum Master?

- A. Participating as an equal member of the Scrum team while acting as a leader who serves.
- B. Assigning action items for team members to complete during the Sprint.
- C. Communicating the development team's answers to stakeholder questions through transcription, email or direct communication.
- D. Taking dictation during the Daily Scrum and reporting discussions that take place during the Daily Scrum back to management.
- E. Scheduling meetings and booking conference rooms for the team.

The correct answer is A.

The Scrum Guide says the Scrum Master is a leader who serves. Note that it does not say the Scrum Master is a 'servant-leader.'

'Leader who serves' is the new terminology.

It is not the job of the Scrum Master to perform secretarial duties for the Scrum Team, nor is it the job of the Scrum Master to be a liaison between management and the developers.

The job of the Scrum Master is to:

- Act as a leader first and a servant to the team second
- Coach and facilitate the Scrum Team and the organization on the application of Scrum
- Ensure the rules of Scrum are observed and coach around instances when they are violated
- Cause the removal of impediments (*cause* the removal, not actually *remove*)
- Help to create a psychologically safe space for everyone involved

A Scrum Master never expected to:

- Assigns tasks
- Disciplines developers
- Run the Daily Scrum
- Select Product Backlog items
- Cancel a Sprint (Only a PO can)
- Override the decisions of the developers

Coach, facilitate and behave as a leader who serves are the key responsibilities of the SM.

## Answer 12

Which of the following Scrum events are timeboxed to limit the amount of time spent on them?

- A. Standup Meetings.
- B. The Release Sprint.
- C. The Sprint Retrospective.
- D. Sprint Planning.
- E. Stakeholder Meetings.
- F. Sprint Zero.
- G. The Daily Scrum.

The correct answers are C, D and G.

The Sprint Retrospective is a timeboxed event that occurs at the end of each Sprint and allows the team to reflect on their processes and identify areas for improvement. The Sprint Retrospective is timeboxed to a maximum of 3 hours.

Sprint Planning is also timeboxed and occurs at the beginning of each Sprint, allowing the team to plan and prioritize their work for the upcoming Sprint. Sprint Planning is timeboxed to a maximum of 8 hours.

The Daily Scrum is another timeboxed event that occurs daily during the Sprint and allows the team to synchronize their work and plan for the day. The Daily Scrum is timeboxed to a maximum of 15 minutes.

The Sprint Review is also timeboxed to a maximum of 4 hours, although the Sprint Review was not mentioned in this question.

Options A, B, E, and F are not correct because they are not Scrum events. On any Scrum exam, references to standup meetings, release Sprints, stakeholder meetings and Sprint Zero will likely be red herrings designed to trick you into selecting an incorrect answer. These topics are never mentioned at all in the Scrum Guide.

## Answer 13

Which of the following are key responsibilities of the Scrum Master on a Scrum Team?

- A. The Scrum Master helps employees and stakeholders understand and enact an empirical and lean approach to solving complex problems.
- B. The Scrum Master assigns Product Backlog items to developers when they are idle.
- C. The Scrum Master helps those outside the Scrum team understand Scrum and how to interact with teams that employ the Scrum framework.
- D. The Scrum Master ensures that all Scrum events are kept within their timebox.
- E. The Scrum Master demos to stakeholders the features that have met the Definition of Done during the Sprint Review.

The correct answers are A, C and D.

Key responsibilities of the Scrum Master include:

- Causing the removing obstacles the team encounters
- Protecting the Scrum Team from external interruptions
- Helping the entire organization enact a lean and empirical approach to problem solving.

Additionally the Scrum Master ensures that the Scrum process is being followed and encourages self-organization and cross-functionality among the Scrum Team members. The Scrum Master should not be seen as a police officer or trailer-park supervisor enforcing rules and commanding others to act. They Scrum Master coaches, facilitates and acts as a leader who serves.

However, there are rules in the Scrum Guide that do need to be respected, one of which is the timebox for Scrum events. There aren't a lot of rules in the Scrum Guide, but when the few rules that exist are violated, it's the job of the Scrum Master to coach the team on the importance on respecting the ethos of Scrum.

## Answer 14

For an Increment of work to be demonstrated to the stakeholders at the Sprint Review, it must be:

- A. Valuable
- B. Paid for
- C. Peer reviewed
- D. Usable
- E. Compatible with past Increments

The correct answers are A, D and E.

An Increment must be valuable, usable, additive and compatible with all past increments.

## Answer 15

The Scrum Guide very briefly mentions burn-down and burn-up charts.

What does a trend line displayed on a burndown chart indicate about the team's progress?

- A. How quickly the team is 'burning through' money budgeted for the project.
- B. How quickly other Scrum teams on the same project are 'burning through' Product Backlog items compared to your team.
- C. The amount of work remaining plotted against time, which predicts when the project will finish if nothing changes with the Product Owner, Product Backlog items or the development team.
- D. How quickly the project will be completed if Product Backlog items are slowly removed by the Product Owner.

The correct answer is C.

Don't worry too much about burn-down charts, as they are only briefly mentioned in the Scrum Guide.

"Various practices exist to forecast progress, like burn-downs, burn-ups, or cumulative flows. While proven useful, these do not replace the importance of empiricism." - The Scrum Guide.

A trend line displayed on a release burndown chart indicates the progress of a project over time. The release burndown chart is a visual representation of the amount of work remaining to complete a project, plotted against time. The generated trend line represents the expected progress of the project if it were to continue at the same rate as it has been progressing up to that point in time.

It is important to note that the trend line is a projection based on past performance, and it may not accurately predict the future progress of the project. It is a predictor, but it is not an *empirical* measure. Empirical measures are preferred in Scrum.

## Answer 16

When should new developers be added to a Scrum team?

- A. During Sprint Zero the team size should be set for the duration of the project.
- B. After the current Sprint ends but before the next Sprint begins.
- C. When adding a new developer will help development continue at a sustainable pace.
- D. When the Project Manager provides enough budget money for a new developer.

The correct answer is C.

A new developer can be added to a project at any point in time. There is no rule barring a developer from joining a Scrum team during Sprint Planning, the Sprint Retrospective or even half-way through a Sprint.

Sustainable development is an important Agile principle. Developers should always be challenged and motivated, but they should never be overworked. If a new developer needs to be added to the team to maintain sustainable development, then add them.

Just keep in mind that onboarding will sometimes reduce the team's productivity in the short-term, as training and orientation for the new hire will likely occupy some of your developer's time.

## Answer 17

The project has kicked off and the first Sprint was a great success.

According to the Scrum Guide, when does the second Sprint begin?

- A. Immediately after the Sprint Review for the first Sprint is completed.
- B. Immediately after Sprint Planning for the second Sprint is completed.
- C. The second Sprint begins immediately after the first Sprint is finished.
- D. When the Scrum Master declares the start of the new Sprint in JIRA.

The correct answer is C.

All Scrum Events happen with a Sprint. Nothing happens 'outside of a Sprint'.

When the Sprint Retrospective finishes, the current Sprint ends. The next Sprint begins immediately after.

## Answer 18

The development team can't agree whether to use Java or Python to develop microservices for their current project.

How can the Scrum Master help the Scrum Team resolve an internal disagreement about whether to build in Java or Python?

- A. Have the developers consult an external, impartial expert on the topic and agree to go with the external expert's decision.
- B. Have the development team argue both sides to the Scrum Master and have the Scrum Master come to a final, impartial decision.
- C. Use coaching techniques like reflective listening and visualization to help guide the entire development team towards building a consensus.
- D. Teach the development team about collaborative techniques to build consensus, such as actively listening to others and paraphrasing contrary positions.

The correct answers are C and D.

Coaching and teaching are important parts of the Scrum Master's role as a leader who serves.

Active listening and open questioning are important skills in effective communication, particularly in Scrum where collaboration and continuous feedback are key. However, there are other techniques and approaches that can also be useful in communication, including:

Reflective listening: This involves reflecting back what the speaker has said to demonstrate that you understand their message. For example, "So what I hear you saying is that you're concerned about the timeline for this project."

Summarizing: This involves summarizing what has been said to ensure that everyone is on the same page. For example, "Let me make sure I understand - you're saying that we need to focus on improving the user experience for this feature."

Paraphrasing: This involves restating what has been said in your own words to show that you understand and to encourage clarification. For example, "If I'm understanding you correctly, you're suggesting that we approach this problem from a different angle."

Nonverbal communication: This includes using body language, eye contact, and facial expressions to show that you are engaged and paying attention.

Visualization: This involves using diagrams, sketches, or other visual aids to help communicate ideas and concepts.

Silence: Sometimes, allowing a pause in the conversation can give the speaker time to gather their thoughts and provide a more thoughtful response.

A good Scrum Master will bring all of these coaching and facilitation techniques to their Scrum Teams and empower their teams to use them in order to effectively and independently solve problems and resolve conflicts.

It's not the Scrum Master's job to resolve conflicts or make decisions for the team. Instead, the Scrum Master will coach, facilitate and provide conflict resolution tools to the team so that they can resolve issues internally and on their own.

The Scrum Guide doesn't say the Scrum Master should remove obstacles for the team. Instead, it says the Scrum master should *cause* the removal of impediments for the team. Coaching and facilitating the Scrum Team is one of the ways they do this.

## Answer 19

When is a Product Backlog item (PBI) considered fully complete and ready for release?

- A. A Product Backlog item is complete when a Quality Assurance team verifies that the work passes all acceptance criteria, making it ready for release
- B. A Product Backlog item is complete when all work in the Sprint Backlog related to the item is finished, making it ready for release
- C. A Product Backlog item is complete when it meets the Definition of Done and further Increments of work on it are no longer possible, making it ready for release.
- D. Product Backlog items are never officially complete until the Product Goal is achieved.

The correct answer is C.

A Product Backlog item is complete when all of its features are completed, meet the Definition of Done, and it is not possible to create any further Increments out of it.

A Product Backlog item is not necessarily associated with a single Increment of Work. A PBI may get decomposed into multiple, smaller pieces and work. Work on a decomposed PBI can last multiple Sprints, with each Sprint producing an Increment that meets a Definition of Done for a given facet of that Product Backlog item.

That's why PBIs are decomposed by the development team - to break them up into manageable pieces of work that can be spread out over time and be planned out more accurately.

## Answer 20

Who on the Scrum Team has the responsibility of ordering the Product Backlog?

- A. The Product Owner orders the Product Backlog, with the items that deliver the most value to the stakeholders at the top.
- B. The Development Team orders the Product Backlog, with items listed chronologically according to what the development team should logically build first.
- C. The Scrum Master orders the Product Backlog, with the items that the Product Owner says will deliver the most value to the stakeholders at the top.
- D. The Product Owner prioritizes the Product Backlog, with PBIs with the highest story-points assigned to them first.

The correct answer is A.

In the Scrum framework, the Product Owner is responsible for creating and maintaining the Product Backlog, which is an ordered list of features, enhancements, and fixes that the Scrum Team will work on in order to deliver a product Increment.

The Product Owner orders the items in the Product Backlog based on a variety of factors that relate to how the items bring value to the business. Generally speaking, items that are believed to provide the most value to the product's users and customers are placed at the top of the list, and those that are of lower priority are placed towards the bottom.

The Scrum Guide says the Product Backlog gets *ordered*, not *prioritized*. Many seasoned Scrum Masters are surprised to learn that the 2020 Scrum Guide never once uses the word *prioritized*.

## Answer 21

What are the two primary responsibilities of a new Product Owner when taking ownership of a product that is currently under development with a long history of multiple, successful Sprints?

- A. Ensuring that the development teams know which Product Backlog items provide the greatest value and should be developed first
- B. Describing and fully decomposing product features in Use Case and story form.
- C. Providing detailed development specs and guidelines to the development team.
- D. Interacting with stakeholders to learn how to represent their needs in the Product Backlog.
- E. Creating both functional and non-functional tests to validate the Increment meets the Definition of Done.

The correct answers are A and D.

The Product Owner must order the Product Backlog so that developers know which items provide the most value. If possible, the Product Owner will negotiate with the team to ensure these items get developed first.

Also, according to the Scrum Guide, "The Product Owner may represent the needs of many stakeholders in the Product Backlog." So interacting with and collaborating with stakeholders is important.

## Answer 22

Under what circumstances might the decision be made to prematurely terminate a Sprint?

- A. When a self-managed team of Scrum Developers realizes the work committed to during Sprint planning cannot possibly be delivered by the end of the Sprint.
- B. When the Sprint Goal no longer makes sense and becomes obsolete.
- C. When the Scrum Development team's work needs to be redirected to new opportunities.
- D. When the Product Owner reprioritized the Product Backlog and PBIs currently under development are no longer priorities.

The correct answer is B.

When the Sprint Goal becomes obsolete the Product Owner can cancel the Sprint.

In Scrum, the Sprint Goal is a short statement that describes what the Development Team intends to achieve during the Sprint. It provides focus and direction to the team and helps to ensure that all members are working towards a common objective. The Sprint Goal is set during Sprint Planning and remains unchanged throughout the Sprint.

If the Sprint Goal becomes obsolete, it means that the objective that the Scrum Developers set for the Sprint is no longer relevant. This can happen for a variety of reasons, such as changes in the business environment or new information that makes the Sprint Goal irrelevant. Essentially, something happened that makes working towards the Sprint Goal a complete waste of time.

When the Sprint Goal becomes obsolete, the Scrum framework allows for the Sprint to be cancelled by the Product Owner.

It is important to note that cancelling a Sprint is not a decision that should be taken lightly. If Sprints are cancelled frequently, it can indicate deeper problems with the project that need to be addressed.

## Answer 23

Corporate stakeholders are very busy, as multiple projects are going online this quarter.

The stakeholders have asked to attend every-other Sprint Review to minimize the number of meetings on their calendars. How do you as a Scrum Master respond?

- A. Coach the stakeholders and the team on how to make the meetings more productive and positive.
- B. Agree with the request of the stakeholders, as reducing meetings is in line with Agile principles.
- C. Educate stakeholders on the importance of Sprint Reviews and work to facilitate more engaging Sprint Reviews.
- D. Consult privately with the Product Owner and see if they will agree to fewer meetings.

The correct answers are A and C.

The Scrum Master should use this opportunity to explain the importance of a Sprint Review and the value the team receives from the stakeholders participating in it.

Attendance at Scrum events is not optional, nor is running Scrum events as planned. Stakeholders must participate in the Sprint Review.

As the Scrum Guide states: "The Scrum framework is immutable. While implementing only parts of Scrum is possible, the result is not Scrum. Scrum exists only in its entirety"

**Answer 24**

As the Scrum Master, it is clear to you that the Sprint Goal is not going to be achieved if the Sprint Backlog isn't adjusted and new Product Backlog items aren't added.

In Scrum, who has the right to add new Product Backlog items (PBIs) to the Sprint Backlog?

- A. The Product Owner prioritizes Product Backlog items and has the authority to add PBIs to the Sprint Backlog.
- B. The development team owns the Sprint Backlog and can add items to it at any time during the Sprint.
- C. Both the Scrum Master and the developers can add to the Sprint Backlog if changes need to be made to keep the Sprint Goal in focus.
- D. Anyone on the Scrum Team can add or remove items from the Sprint Backlog.

The correct answer is B.

The development team has full control over the Sprint Backlog. Only the development team has the authority to add or remove PBIs selected for the Sprint Backlog.

Anyone can discuss the Sprint Backlog with the development team and relay any concerns to them, but it is the development team that has the final say over what is added or removed.

If anyone thinks the Sprint can be improved through the addition or deletion of the Sprint Backlog, they are welcome to discuss those thoughts with the developers. Whether they can convince the developers to act on those discussions is another story.

Developers own the Sprint Backlog.

**Answer 25**

The Product Owner often has no idea how complicated or complex a Product Backlog item will be to build.

Whose responsibility is it to decompose long, epic, Product Backlog items into easily digestible work items that should take no more than a day to complete?

- A. The Business Analysts.
- B. The Product Owner.
- C. The Scrum Master.
- D. The Team Lead.
- E. The Developers on the team.

The correct answer is E.

Breaking Product Backlog items into smaller work items is the job of the developers on the team.

"For each selected Product Backlog item, the Developers plan the work necessary to create an Increment that meets the Definition of Done.

This is often done by decomposing Product Backlog items into smaller work items of one day or less. How this is done is at the sole discretion of the Scrum Developers.

Nobody else tells them how to turn Product Backlog items into Increments of value."

## Answer 26

Scrum doesn't work if there isn't buy-in from the entire organization, including management.

How do members of the management group in an organization that sponsors a development project support a Scrum Team?

- A. The organization empowers the Scrum Team to manage their own work.
- B. The organization responds quickly to daily status updates from the Scrum Master.
- C. The organization must provide Scrum Teams with resources that allow for continual improvement.
- D. The organization fully respects and trusts the Product Owner's decisions.

The correct answers are A, C and D.

Status updates are antithetical to the Scrum Framework. Scrum provides other artifacts and mechanisms to allow for transparency and openness into the progress of the Scrum Team.

Providing sufficient input and resources for the Scrum Team to improve, allowing the Scrum Team to manage their own work, and respecting the decisions of the Product Owner are three concrete ways an organization can support teams that practice Scrum.

## Answer 27

The Scrum team needs to hire an outside specialist, external to the Scrum team, to deliver wireframes needed in order to complete a Product Backlog item and meet the Definition of Done. Some of the developers have complained to you, the Scrum Master, in confidence about the outside specialist.

What are your main concerns in this situation?

- A. You are concerned that the external dependency may impact software development velocity.
- B. You are concerned that team members are coming to you with complaints rather than figuring out a solution themselves within the team.
- C. You are concerned about the ability of the development team to produce a fully integrated Increment when meeting the Definition of Done depends on work produced by someone outside of the Scrum Team.
- D. You are worried that there will be software developers sitting idle while work is outsourced.

The correct answers are B and C.

The Scrum Team should have all the skills required to meet the Definition of Done (DoD). If the DoD depends on work performed outside of the team, there is no control over it and it jeopardizes the team's ability to create a full Increment.

The development team should also be self-organized and self-managed, which implies that they should be able to discuss and resolve concerns internally. Openness and transparency may be lacking if developers are complaining to the Scrum Master in secrecy.

The Scrum Master should work hard to make the workplace a psychologically safe space where developers feel free to discuss their fears and concerns openly and honestly with each other. Secrecy is not consistent with the Scrum pillar of transparency.

## Answer 28

Which of the following is *not* true of the Daily Scrum?

- A. The Daily Scrum is used to identify impediments.
- B. The Daily Scrum eliminates the need for other meetings.
- C. The Daily Scrum is timeboxed to 30 minutes.
- D. The Daily Scrum encourages quick decision-making.

The correct answer is C.

The Daily Scrum is timeboxed to 15 minutes, not 30 minutes.

From the Scrum Guide:

"The Daily Scrum is a 15-minute event for the Developers of the Scrum Team. Daily Scrums:

- Improve communications
- Identify impediments
- Promote quick decision-making
- Consequently eliminate the need for other meetings."

## Answer 29

What are the skills and capabilities the developers on the Scrum Team should have in order to ensure effective performance throughout the Sprint?

- A. The ability to decompose Product Backlog items (PBIs) and progressively create Increments of work until a functional release is possible.
- B. The ability to perform all of the core software development work, with the exception of any specialized testing that would require additional tools and staging environments.
- C. The ability to complete a development project within budget and within the timeline agreed upon with the Product Owner.
- D. The ability to take direction from the Scrum Master and complete Product Backlog items assigned to them by the Product Owner.

The correct answer is A.

The Scrum Developers must be able to decompose a PBI and create as many Increments of work that are necessary to complete the PBI and make the item ready for potential release.

A Product Backlog item (PBI) is often a very *high-level* description of a feature or a function. The Product Owner will know what they want, but they won't always know how to 'get there' or describe it in sufficient detail for the developer to fully understand what is needed. This requires the developers to work with the Product Owner to decompose the Product Backlog item into smaller, constituent parts that are easier to estimate, plan and build.

For example, maybe a Product Owner wants to build a house. They are unlikely to have anything in the Product Backlog about pouring a foundation or connecting to the main sewage line. Those are details the Scrum Developers will have to flush out as they decompose the Product Backlog items.

Decomposing a PBI means breaking it down into smaller, more manageable pieces of work that can

be completed by the Scrum Developers during a Sprint. The goal is to decompose Product Backlog items down into units of work that can be completed in a single day.

The process of decomposing a PBI involves analyzing it in more detail, identifying sub-tasks, dependencies and other factors involved, and then creating smaller Product Backlog items that can be individually prioritized, estimated, and worked on by the team. This allows the team to better understand the requirements of the PBI and the work involved, which in turn helps with planning and delivering the work effectively.

Decomposition is an ongoing process that takes place throughout every Sprint.

As the team progresses through the Sprint, they may discover additional details or dependencies that require further decomposition of Product Backlog items, or they may need to adjust their plan based on feedback from stakeholders or changes in the product's requirements. By continually decomposing Product Backlog items, the team can ensure they have a clear understanding of the work involved, which helps the team prioritize, estimate, and deliver Product Backlog items more transparently.

## Answer 30

The development team bit off more than it could chew during Sprint Planning.

They've just told you, the Scrum Master, that there's no way they are even going to come close to completing the Sprint Backlog.

What should you, as the Scrum Master, do?

- A. Change the Definition of Done and negotiate the quality standard with the Product Owner to ensure all Sprint Backlog items are completed before the end of the Sprint.
- B. While the Sprint is ongoing, have the Product Owner and the developers negotiate the scope of the work being done so as not to endanger the Sprint Goal.
- C. Scale out and assign some of the Sprint Backlog items to an external Scrum Team.
- D. Inform the Product Owner prior to the Sprint Review so the stakeholders won't be disappointed.

The correct answer is B.

It's not unusual to have a Sprint go sideways. When that happens, the Scrum Team has to adapt.

It's acceptable to clarify and negotiate scope during a Sprint. In fact, it would be difficult to imagine a Sprint where the PO and the developers don't spend some time negotiating the scope of the Sprint.

After all, some tasks will get completed faster than expected. Other tasks might take longer. When

that happens, developers should sit down with the PO and figure out what items they might want to add to the Sprint Backlog or which items they might want to remove.

Open discussions like this will occur naturally on any healthy product development team.

According to the Scrum Guide:

"During the Sprint:

- No changes are made that would endanger the Sprint Goal;
- Quality does not decrease;
- The Product Backlog is refined as needed; and,
- Scope may be clarified and renegotiated with the Product Owner as more is learned."

## Answer 31

What should be done with a Product Backlog item (PBI) that did not meet the Scrum Team's agreed upon Definition of Done by the end of a Sprint?

- A. Return the item to the Product Backlog.
- B. With the consent of the Product Owner and stakeholders, the PBI can be approved and released.
- C. Automatically add the unfinished PBI to the next Sprint's backlog.
- D. Make sure the team knows that work on this PBI will not be included in the current Sprint's Increment.

The correct answers are A and D.

Incomplete PBIs not completed in a given Sprint are returned to the Product Backlog.

They are not automatically added to the next Sprint, as priorities may have changed since the prior round of Sprint Planning took place.

If a PBI is not completed, it definitely should not be added to the Increment. The Increment is only for completed work that meets the DoD.

The Product Backlog is the ordered list of features, requirements, and other work items that the development team plans to complete over the course of the project. During each Sprint, the team selects a subset of items from the Product Backlog to work on, creating a Sprint Backlog.

One of the key principles of Scrum is that the team commits to completing the work it selects for the Sprint Backlog within the Sprint. However, if any Product Backlog items are not completed by the end of the Sprint, they are not automatically added to the next Sprint Backlog. Instead, they are returned to the Product Backlog for reordering and consideration for future Sprints.

## Answer 32

Teams sometimes struggle to come up with items to discuss during the Sprint Retrospective.

Which of the following topics and issues would be worthwhile for a Scrum Team to discuss during a Sprint Retrospective?

- A. Whether the Definition of Done needs to be updated.
- B. Whether the length of the Sprint needs adjusting.
- C. How to better decompose Product Backlog items during Sprint Planning.
- D. How to improve communication between on-site and remote workers.
- E. All of the above.

The correct answer is E.

All of these topics are commonly discussed in the Sprint Retrospective.

From the Scrum Guide: "[During the Sprint Retrospective,] the Scrum Team inspects how the last Sprint went with regards to:

- Individuals
- Interactions
- Processes and Tools
- The Definition of Done.

Inspected elements often vary with the domain of work."

Remember that only members of the Scrum Team are invited to the Sprint Retrospective. This is in stark contrast to the Sprint Review where anyone with a stake in the development of the product can attend.

## Answer 33

Which of the following actions should the Scrum Master never do?

- A. Tell the Scrum Developers how to manage their workloads.
- B. Extend the length of the Sprint.
- C. End the Sprint early.
- D. Coach the Development Team to ensure the Daily Scrum is timeboxed to 15 minutes.
- E. Reprimand developers who underperform.

The correct answers are A, B, C and E.

The Scrum Master is a leader who serves the team and as such, should never *tell* the developers what to do. The Scrum Master coaches and facilitates and allows developers to discover solutions and solve problems themselves.

Furthermore, the Scrum Framework does not allow the length of a Sprint to be extended, and only the Product Owner can end a Sprint early.

The Scrum Master does coach teams about the Scrum Framework, so coaching a team to ensure the 15 minute Daily Scrum timebox is not exceeded is a good use of the Scrum Master's time.

## Answer 34

Sprint Planning is the first of the four timeboxed events to happen during the Sprint.

According to the Scrum Guide, which topics are to be addressed by the Scrum Team during Sprint Planning?

- A. How on Earth is the work chosen for the Sprint going to get done?
- B. Why is this a valuable Sprint?
- C. What do you think can get done before the end of this Sprint?
- D. Why were certain Sprint Backlog items not completed in the prior Sprint?
- E. When will infrastructure required for deployment will be ready?

The correct answers are A, B and C.

According to the Scrum Guide, a Sprint Planning meeting addresses these three questions:

- Why is this Sprint valuable?
- What can be done this Sprint?
- How will the chosen work get done?

Always remember that *why*, *what* and *how* are the questions to be answered before the end of the Sprint Planning event.

## Answer 35

According to the Scrum Guide, which two of the following are true about the Product Owner?

- A. The Product Owner may delegate the creation of Product Backlog items to an associate.
- B. The Product Owner may be a committee or collection of collaborating individuals.
- C. The organization must fully respect and trust the independent decisions of the Product Owner.
- D. The Product Owner should coach the developers in self-management and cross-functionality.
- E. The Product Owner must attend and participate in the Daily Scrum.

The correct answers are A and C.

The Product Owner is allowed to delegate certain aspects of their job to others on the team, as the Scrum Guide indicates:

"The Product Owner ... may delegate the responsibility to others. Regardless, the Product Owner remains accountable. For Product Owners to succeed, the entire organization must respect their decisions. These decisions are visible in the content and ordering of the Product Backlog, and through the inspectable Increment at the Sprint Review. The Product Owner is one person, not a committee."

The organization must also fully trust the decisions of the Product Owner. The PO must be trusted to make quick decisions about the product so that developers can get fast responses and quickly resolve issues.

## Answer 36

The team's developers are amazing! Maybe too amazing?

It's not even half-way through the Sprint and the developers are telling you, the Scrum Master, that they have not selected enough items for the Sprint. Everything is practically done, which will result in developers being underutilized with nothing to do for the rest of the Sprint.

What should the Scrum Master do?

- A. End the Sprint and reconvene a new Sprint Planning meeting.
- B. Have the Product Owner add new Product Backlog items to the Sprint Backlog.
- C. Create a more rigorous Definition of Done so that quality will improve and it will require more work to complete the items currently in the Sprint Backlog

- D. Have the developers adapt their Sprint Plan, preferably with the help of the Product Owner, and add additional, high-value Product Backlog items to the Sprint Backlog.

The correct answer is D.

If developers complete Sprint Backlog items quickly and become underutilized, they should adapt and add new Product Backlog items to the Sprint Backlog.

First and foremost, Scrum is empirical. But Scrum is also pragmatic. Scrum's rules allow for simple solutions to problems that arise, and approaching question on the Scrum Master certification exam pragmatically will help you score some easy points.

If developers didn't select enough work, then let them select more work. It's really that simple.

Never overcomplicate Scrum.

## Answer 37

During every Sprint, the team attempts to create at least one Increment to present to stakeholders during the Sprint Review.

Why might a Product Owner refuse to immediately release an Increment into production after a successful Sprint Review?

- A. Increments are automatically released into production so it is not possible to refuse a release.
- B. The Product Owner needs confirmation from the Scrum Master in order to release an Increment into production.
- C. The immediate timing of the release does not make sense for customers or stakeholders.
- D. The Product Owner never refuses to release an Increment into production.

The correct answer is C.

If the timing of the release of an Increment is not in the best interest of users or stakeholders, the Product Owner does not need to release it immediately.

For example, a Christmas Theme for the website might be completed in late October. You wouldn't publish the website's Christmas theme two months before Christmas.

Increments don't need to be released or published immediately. They can be released when they make the most sense for the organization, stakeholders and customers.

## Answer 38

The developers are new to Scrum and they've asked you, an experienced Scrum Master, who should start, lead and manage the Daily Scrum. How do you respond?

- A. Explain that the most senior developer on the team should start and lead the Daily Scrum
- B. Explain that the most senior developer on the team leads the Daily Scrum while the most junior developer speaks first
- C. Explain that the team lead leads the Daily Scrum while the most senior developer speaks first
- D. Explain how the developers can choose whatever structure they want for the Daily Scrum, so who starts it is up to them.
- E. Explain that Scrum is hierarchy-less and it doesn't require any specific individual to 'lead' during the Daily Scrum.

The correct answers are D and E.

It is up to the developers to decide how to run their Daily Scrum, which includes decisions about who starts it and how it proceeds.

"The Developers can select whatever structure and techniques they want, as long as their Daily Scrum focuses on progress toward the Sprint Goal and produces an actionable plan for the next day of work. This creates focus and improves self-management."

Furthermore, there are no 'team leads' or 'junior developers' in the eyes of Scrum.

According to the Scrum Guide: "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."

## Answer 39

You are building a limo for a head of state, and security and protection of the occupant are two important, non-functional requirements that must be prioritized at every step in the development process.

What is the best way to ensure security and protection of the occupant are prioritized?

- A. Outsource security and protection concerns to an external third party that specializes in these areas.
- B. Add a Sprint prior to the release of the car that deals exclusively with security and protection.
- C. Build a special sub-team on the development team that deals exclusively with security and protection of the occupant.
- D. Have the Product Owner add the features that pertain to security and protection of the occupant to the Product Backlog.
- E. Add concerns related to the security and protection of the occupant to the Definition of Done.

The correct answers are D and E.

We don't outsource work in Scrum, nor do we create sub-teams.

From the Scrum Guide: "Scrum Teams are cross-functional, meaning the members have all the skills necessary to create value each Sprint. 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."

In Scrum, we deal with non-functional requirements by either adding Product Backlog items that address these concerns, or we add non-functional criteria to the Definition of Done.

## Answer 40

Which of the following is true about the Product Owner's attendance at the Daily Scrum?

- B. The Product Owner must always attend the Daily Scrum to help remove any impediments that exist that jeopardize the Sprint Goal
- B. The Product Owner must always attend the Daily Scrum so they can quickly answer any questions the Scrum Developers have about items they are working on.
- C. The Product Owner must always attend the Daily Scrum to present the stakeholder's point of view.
- D. The Product Owner doesn't need to attend the Daily Scrum.

The correct answer is D.

The Product Owner does not participate in the Daily Scrum, nor are they required to attend it. They only participate in the Daily Scrum at the behest of the Scrum Developers. For example, the developers might phone the Product Owner during a Daily Scrum to get clarification on a topic, or perhaps send them a text message. But the Product Owner is not required to attend the Daily Scrum, and they only participate if the developers need their input.

In fact, not even the Scrum Master is required to attend the Daily Scrum, and they are not allowed to actively participate in it. Like the Product Owner, a Scrum Master only participates in the Daily Scrum at the behest of the developers.

The Daily Scrum is for the developers.

Unless the Scrum Master or Product Owner is actually doing development for some reason and acting as a developer working on a backlog item, then they should not participate in the Daily Scrum. The Daily Scrum is only for the developers.

Congratulations!

You got to the end of this book. You've finished reading my Scrum Master Certification Guide!

And you're probably wondering 'what now?' What's next?

I'll tell you what's next.

What's next is that you need to:

- Go onto the scrum.org site
- Purchase an exam voucher
- Schedule the exam
- Take the exam

## Put your fears aside

This book provides more than enough detail to help you pass the Scrum Alliance Scrum Master exam too, although I personally believe the scrum.org cert is better.

Just pick your poison and schedule the exam.

Now I know what you're thinking.

- You're worried that you're not ready. **Doesn't matter. Schedule the exam.**
- You're worried you need to study more. **Doesn't matter. Schedule the exam.**
- You think you should take another course or read another book. **Don't do it. Schedule the exam.**

## Some unorthodox advice

And I'd also offer up this unorthodox advice. Take the exam with the expectation that you're going to fail.

The fact is, most people fail the first time, and it's rarely due to them not being knowledgeable enough.

The time constraints on the exam are savage. You have to answer a question every minute.

Until you actually take the exam in a live setting, you really have no appreciation for how stressful that is.

With a passing score above 80%, many people fail simply because they don't answer all the questions in time.

## Sign up and do a dry run

Other people just get flustered in the exam environment.

It's unfamiliar and the stakes are high when you take the test, so fumbling around over how to mark a question to view it later or how to move back to previously asked questions can throw you off your game.

The exam interface isn't that complicated, so after you've taken the test once you'll be an expert with it.

But the first time? It can be intimidating, and that intimidation can mess you up.

## Getting into their head

There's also a rhythm to the questions on the exam.

It's almost like you have to get ten or fifteen questions in to understand what was in the head of the people who designed the test.

There's just a certain way questions are asked, and the way questions are worded that takes a little while to get comfortable with.

I certainly try to replicate that rhythm in the practice tests as much as I can, but I can't do it perfectly.

Once you tune into the rhythm, answering questions becomes easier; but the first time you write the exam you might get a question or two incorrect just because the wording of the question might feel awkward. With a really high passing score, one or two missed questions can be the difference between a pass or a fail.

## An empirical evaluation

And you know what?

You might fail because you're not prepared or because there are certain topics you are weak on. Finding that out is a good thing, not a bad thing.

After you take the Scrum Master certification exam, your score is broken down into 13 different categories. This gives you immediate feedback on where you need to concentrate your studies.

Furthermore, when you do take the exam, you will inevitably encounter some terms you may not have heard before, or perhaps you had not paid enough attention to.

But after taking the exam, every time you study and that term appears, your radar will go off and you'll pay attention to that term more than ever.

Your focus will be heightened and your time spent studying will be more productive than ever.

## Sample PSM I Assessment Result

Here's a sample report from a student of mine who recently passed the exam:

Scrum.org PSM I Assessment Result	Focus Area
Score	Facilitation
100%	Coaching and Mentoring
100%	Artifacts
100%	Events
100%	Self-Managing Teams
90.9%	Scrum Team
100%	Scrum Values
100%	Empiricism
100%	Product Backlog Management
100%	Done
88.9%	Forecasting and Release Planning
100%	Stakeholders and Customers
100%	Product value

## Zero risk. Massive reward.

There's no penalty to taking the Scrum Master exam and failing it. I know lots of Scrum experts who failed the exam the first time. It's common.

But at the same time, what an awesome opportunity to find out what the exam is like, what the questions feel like, how the exam software works and get immediate and targeted feedback about where you need to concentrate your studies.

In the world of Scrum, we talk about the importance of empirical research and data. What's more empirical than taking the exam and examining your results.

Sure, it costs \$100.

That's a sunk cost, and \$100 isn't a trivial amount of money. But it's also part of an investment you're making in yourself, and from that perspective, I think it's more than worth it.

I mean, I've seen people pay over \$200 to purchase a set of online practice exams. Why not just take the actual exam instead?

## You may surprise yourself...

Now here's the thing.

I want you to go in and take the exam with the full expectation that you'll fail the first time.

But you know what often happens? Often people pass.

Sometimes people who feel they are completely unready will score a perfect 100%.

## You might be over-preparing?

The majority of people who take the Scrum Master certification exam fail on their first attempt.

But at the same time, the majority of people who pass have over prepared.

I've seen hundreds of students push the exam off for months, doing practice exams and re-reading the Scrum Guide until they've got every paragraph memorized. Don't do that. Your time is far too valuable.

How do you know if you're over prepared or underprepared? You take the exam and find out!

## You've got this!

And while it's true the majority of people who take the Scrum Master exam the first time, I will say that statistic is not true about people who have read this certification guide and taken its practice tests seriously.

If you can go through all the sample questions in this book and both answer them correctly and be able to personally articulate why a given answer is correct, then there is a very good chance you'll pass the exam on the first try. Lots of people have.

Oh, and when you do pass, I want to know about it!

I'm very active on LinkedIn and Twitter. So find me there!

## I want to share in your success

Tag me in a Twitter post with a photo of the book and let the world know that you passed the exam.

I really want to know about your success, and I know others who are struggling themselves will be inspired by seeing you achieve their goals.

Let's inspire people together!

Be @Scrumtuous and Follow @Xennial on Twitter

## A few people I need to thank...

Everyone says that social media is so negative and divisive, but the fact is, I don't know where I'd be if it wasn't for the ongoing support and encouragement from the friends I've made on Twitter. (And yes, I know it's called X now.)

Xennial Innovations	@xennialinc
Chuck LaPress	charleslapress.com
Michael Lloyd	#DysfunctionMapping
Walker Boh	@nakedMCSE
Bojan	@bojbojbojbojboj
Kris Hultner	@KrisHultner
The Pocket Programmer	@PocketProgram
Gareth	@ggwhiting
Andrew Brown	@andrewbrown
Josh Long	@starbuxman
Ted Neward	@tedneward
Jack Forge	@thejackforge
Open Dev Log	@OpenDevLog
Stoic Michael McGill	@mcgillmd921
Uncle Bob Martin	@unclebobmartin
Cameron McKenzie	@cameronmcnz
Allen Holub	@allenholub
Krishna S	@Firestarter7676
Brian Dordevic	@briandordevic
Kevin McAleer	@kevsmac
Sean Kaye	@SeanDoesLife
Kaje	@Quasicodo
Ben Weddle	@ben_weddle
Eddie Jaoude	@eddiejaoude
Minimalish	@Minimalish_YT
Rami M. Amin	@rmtux
Liz Rowe	@lizroweyt

I know I've forgotten a bunch of people. If you're a long-time supporter and follower, message me on Twitter (@Scrumtuous) so I can correct the mistake. Again, thank you all so much!

Be @Scrumtuous and Follow @Xennial on Twitter

## More to come!

This Scrum Master Certification Guide is just the start!

Follow me on Twitter, head over to my website or subscribe to my newsletter to learn more about:

- My upcoming Product Owner Practice Exams book
- My online Scrum Master Practice Tests with over 350 unique sample questions
- My Discord channel where we talk about all things Scrum
- My online Scrum Master Certification Course

Much of this is all in the development stage, so keep in touch and join me as I build in public!

**Go to [www.scrumtuous.com](http://www.scrumtuous.com) to learn more!**



**@scrumtuous**



Figure 39. Please come find me on Twitter. I'm @Scrumtuous on all platforms!

Be @Scrumtuous and Follow @Xennial on Twitter

---

When the student is *truly* ready, the teacher will disappear.

*Tao Te Ching*

---