

## **Scrum Guide for PSM Level I Exam**

## Table of Contents

Scrum Guide for PSM Level I Exam .....	1
Table of Contents .....	2
1. Definition of Scrum .....	3
2. Scrum Theory.....	4
2.1. Transparency.....	4
2.2. Inspection.....	4
2.3. Adaptation.....	4
3. The Scrum Team.....	5
3.1. The Product Owner .....	5
3.2. The Development Team.....	5
3.2.1. Development Team Size .....	6
3.3. The Scrum Master.....	6
3.3.1. Scrum Master Service to the Product Owner .....	6
3.3.2. Scrum Master Service to the Development Team.....	7
3.3.3. Scrum Master Service to the Organization .....	7
4. Scrum Events .....	8
4.1. The Sprint.....	8
4.2. Sprint Planning.....	9
4.3. Sprint Goal .....	10
4.4. Daily Scrum .....	10
4.5. Sprint Review.....	11
4.6. Sprint Retrospective.....	12
5. Scrum Artifacts .....	13
5.1. Product Backlog.....	13
5.2. Sprint Backlog .....	14
5.3. Increment .....	15
6. Artifact Transparency .....	16
7. Definition of “Done”.....	17
8. Q and A.....	18
9. Glossary .....	28
End of document .....	31

# 1. Definition of Scrum

Scrum is a framework for developing and sustaining complex products.

Scrum (n): A framework within which people can address complex adaptive problems, while productively and creatively delivering products of the highest possible value.

Scrum is:

- Lightweight
- Simple to understand
- Difficult to master

Scrum is a process framework that has been used to manage complex product development since the early 1990s. Scrum is not a process or a technique for building products; rather, it is a framework within which you can employ various processes and techniques. Scrum makes clear the relative efficacy of your product management and development practices so that you can improve.

The Scrum framework consists of Scrum Teams and their associated roles, events, artifacts, and rules. Each component within the framework serves a specific purpose and is essential to Scrum's success and usage.

The rules of Scrum bind together the events, roles, and artifacts, governing the relationships and interaction between them. The rules of Scrum are described throughout the body of this document.

Specific tactics for using the Scrum framework vary and are described elsewhere.

## 2. Scrum Theory

Scrum is founded on empirical process control theory, or **empiricism**. Empiricism asserts that knowledge comes from experience *and* making decisions based on what is known. Scrum employs an iterative, incremental approach to optimize predictability and control risk. Three pillars uphold every implementation of empirical process control: **transparency**, **inspection**, and **adaptation**.

### 2.1. Transparency

Significant aspects of the process must be visible to those responsible for the outcome. Transparency requires those aspects be defined by a common standard so observers share a common understanding of what is being seen.

For example:

- A common language referring to the process must be shared by all participants; and,
- Those performing the work and those accepting the work product must share a common definition of “Done”.

### 2.2. Inspection

Scrum users must frequently inspect Scrum artifacts and progress toward a Sprint Goal to detect undesirable variances. Their inspection should not be so frequent that inspection gets in the way of the work. Inspections are most beneficial when diligently performed by skilled inspectors at the point of work.

### 2.3. Adaptation

If an inspector determines that one or more aspects of a process deviate outside acceptable limits, and that the resulting product will be unacceptable, the process or the material being processed must be adjusted. An adjustment must be made as soon as possible to minimize further deviation.

Scrum prescribes four formal events for inspection and adaptation, as described in the *Scrum Events* section of this document:

- Sprint Planning
- Daily Scrum
- Sprint Review
- Sprint Retrospective

### 3. The Scrum Team

The Scrum Team consists of a **Product Owner**, the **Development Team**, and a **Scrum Master**. Scrum Teams are self-organizing and cross-functional. Self-organizing teams choose how best to accomplish their work, rather than being directed by others outside the team. Cross-functional teams have all competencies needed to accomplish the work without depending on others not part of the team. The team model in Scrum is designed to optimize flexibility, creativity, and productivity.

Scrum Teams deliver products iteratively and incrementally, maximizing opportunities for feedback. Incremental deliveries of “Done” product ensure a potentially useful version of working product is always available.

#### 3.1. The Product Owner

The Product Owner is responsible for maximizing the value of the product and the work of the Development Team. How this is done may vary widely across organizations, Scrum Teams, and individuals.

The Product Owner is the sole person responsible for managing the **Product Backlog**.

Product Backlog management includes:

- Clearly expressing Product Backlog items;
- Ordering the items in the Product Backlog to best achieve goals and missions;
- Optimizing the value of the work the Development Team performs;
- Ensuring that the Product Backlog is visible, transparent, and clear to all, and shows what the Scrum Team will work on next; and,
- Ensuring the Development Team understands items in the Product Backlog to the level needed.

The Product Owner may do the above work, or have the Development Team do it. However, the Product Owner remains accountable.

The Product Owner is one person, not a committee. The Product Owner may represent the desires of a committee in the Product Backlog, but those wanting to change a Product Backlog item’s priority must address the Product Owner.

For the Product Owner to succeed, the entire organization must respect his or her decisions. The Product Owner’s decisions are visible in the content and ordering of the Product Backlog. No one is allowed to tell the Development Team to work from a different set of requirements, and the Development Team isn’t allowed to act on what anyone else says.

#### 3.2. The Development Team

The Development Team consists of professionals who do the work of delivering a potentially releasable Increment of “Done” product at the end of each Sprint. Only members of the Development Team create the Increment.

Development Teams are structured and empowered by the organization to organize and manage their own work. The resulting synergy optimizes the Development Team's overall efficiency and effectiveness.

Development Teams have the following characteristics:

- They are self-organizing. No one (not even the Scrum Master) tells the Development Team how to turn Product Backlog into Increments of potentially releasable functionality;
- Development Teams are cross-functional, with all of the skills as a team necessary to create a product Increment;
- Scrum recognizes no titles for Development Team members other than Developer, regardless of the work being performed by the person; there are no exceptions to this rule;
- Scrum recognizes no sub-teams in the Development Team, regardless of particular domains that need to be addressed like testing or business analysis; there are no exceptions to this rule; and,
- Individual Development Team members may have specialized skills and areas of focus, but accountability belongs to the Development Team as a whole.

### **3.2.1. Development Team Size**

Optimal Development Team size is small enough to remain nimble and large enough to complete significant work within a Sprint. Fewer than three Development Team members decrease interaction and results in smaller productivity gains. Smaller Development Teams may encounter skill constraints during the Sprint, causing the Development Team to be unable to deliver a potentially releasable Increment. Having more than nine members requires too much coordination. Large Development Teams generate too much complexity for an empirical process to manage. The Product Owner and Scrum Master roles are not included in this count unless they are also executing the work of the Sprint Backlog.

## **3.3. The Scrum Master**

The Scrum Master is responsible for ensuring Scrum is understood and enacted. Scrum Masters do this by ensuring that the Scrum Team adheres to Scrum theory, practices, and rules.

The Scrum Master is a servant-leader for the Scrum Team. The Scrum Master helps those outside the Scrum Team understand which of their interactions with the Scrum Team are helpful and which aren't. The Scrum Master helps everyone change these interactions to maximize the value created by the Scrum Team.

### **3.3.1. Scrum Master Service to the Product Owner**

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

- Finding techniques for effective Product Backlog management;

- Helping the Scrum Team understand the need for clear and concise Product Backlog items;
- Understanding product planning in an empirical environment;
- Ensuring the Product Owner knows how to arrange the Product Backlog to maximize value;
- Understanding and practicing agility; and,
- Facilitating Scrum events as requested or needed.

### **3.3.2. Scrum Master Service to the Development Team**

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

- Coaching the Development Team in self-organization and cross-functionality;
- Helping the Development Team to create high-value products;
- Removing impediments to the Development Team's progress;
- Facilitating Scrum events as requested or needed; and,
- Coaching the Development Team in organizational environments in which Scrum is not yet fully adopted and understood.

### **3.3.3. Scrum Master Service to the Organization**

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

- Leading and coaching the organization in its Scrum adoption;
- Planning Scrum implementations within the organization;
- Helping employees and stakeholders understand and enact Scrum and empirical product development;
- Causing change that increases the productivity of the Scrum Team; and,
- Working with other Scrum Masters to increase the effectiveness of the application of Scrum in the organization.

## 4. Scrum Events

Prescribed events are used in Scrum to create regularity and to minimize the need for meetings not defined in Scrum. All events are time-boxed events, such that every event has a maximum duration. Once a Sprint begins, its duration is fixed and cannot be shortened or lengthened. The remaining events may end whenever the purpose of the event is achieved; ensuring an appropriate amount of time is spent without allowing waste in the process. Other than the Sprint itself, which is a container for all other events, each event in Scrum is a formal opportunity to inspect and adapt something. These events are specifically designed to enable critical transparency and inspection. Failure to include any of these events results in reduced transparency and is a lost opportunity to inspect and adapt.

### 4.1. The Sprint

The heart of Scrum is a Sprint, a time-box of one month or less during which a “Done”, useable and potentially releasable product Increment is created. Sprints best have consistent durations throughout a development effort. A new Sprint starts immediately after the conclusion of the previous Sprint.

Sprints contain and consist of the Sprint Planning, Daily Scrums, the development work, the Sprint Review, and the Sprint Retrospective.

During the Sprint:

- No changes are made that would endanger the Sprint Goal;
- Quality goals do not decrease; and,
- Scope may be clarified and re-negotiated between the Product Owner and Development Team as more is learned.

Each Sprint may be considered a project with no more than a one-month horizon. Like projects, Sprints are used to accomplish something. Each Sprint has a definition of what is to be built, a design and flexible plan that will guide building it, the work, and the resultant product.

Sprints are limited to one calendar month. When a Sprint’s horizon is too long the definition of what is being built may change, complexity may rise, and risk may increase. Sprints enable predictability by ensuring inspection and adaptation of progress toward a Sprint Goal at least every calendar month. Sprints also limit risk to one calendar month of cost.

#### 4.1.1. Cancelling a Sprint

A Sprint can be cancelled before the Sprint time-box is over. Only the Product Owner has the authority to cancel the Sprint, although he or she may do so under influence from the stakeholders, the Development Team, or the Scrum Master.

A Sprint would be cancelled if the Sprint Goal becomes obsolete. This might occur if the company changes direction or if market or technology conditions change. In general, a Sprint should be cancelled if it no longer makes sense given the circumstances. But, due to the short duration of Sprints, cancellation rarely makes sense.



When a Sprint is cancelled, any completed and “Done” Product Backlog items are reviewed. If part of the work is potentially releasable, the Product Owner typically accepts it. All incomplete Product Backlog Items are re-estimated and put back on the Product Backlog.

The work done on them depreciates quickly and must be frequently re-estimated. Sprint cancellations consume resources, since everyone has to regroup in another Sprint Planning to start another Sprint. Sprint cancellations are often traumatic to the Scrum Team, and are very uncommon.

## **4.2. Sprint Planning**

The work to be performed in the Sprint is planned at the Sprint Planning. This plan is created by the collaborative work of the entire Scrum Team.

Sprint Planning is time-boxed to a maximum of eight hours for a one-month Sprint.

For shorter Sprints, the event is usually shorter. The Scrum Master ensures that the event takes place and that attendants understand its purpose. The Scrum Master teaches the Scrum Team to keep it within the time-box.

Sprint Planning answers the following:

- What can be delivered in the Increment resulting from the upcoming Sprint?
- How will the work needed to deliver the Increment be achieved?

### **4.2.1. What can be done this Sprint?**

The Development Team works to forecast the functionality that will be developed during the Sprint. The Product Owner discusses the objective that the Sprint should achieve and the Product Backlog items that, if completed in the Sprint, would achieve the Sprint Goal. The entire Scrum Team collaborates on understanding the work of the Sprint.

The input to this meeting is the Product Backlog, the latest product Increment, projected capacity of the Development Team during the Sprint, and past performance of the Development Team. The number of items selected from the Product Backlog for the Sprint is solely up to the Development Team. Only the Development Team can assess what it can accomplish over the upcoming Sprint.

After the Development Team forecasts the Product Backlog items it will deliver in the Sprint, the Scrum Team crafts a Sprint Goal. The Sprint Goal is an objective that will be met within the Sprint through the implementation of the Product Backlog, and it provides guidance to the Development Team on why it is building the Increment.

### **4.2.2. How will the chosen work get done?**

Having set the Sprint Goal and selected the Product Backlog items for the Sprint, the Development Team decides how it will build this functionality into a “Done” product

Increment during the Sprint. The Product Backlog items selected for this Sprint plus the plan for delivering them is called the Sprint Backlog.

The Development Team usually starts by designing the system and the work needed to convert the Product Backlog into a working product Increment. Work may be of varying size, or estimated effort. However, enough work is planned during Sprint Planning for the Development Team to forecast what it believes it can do in the upcoming Sprint. Work planned for the first days of the Sprint by the Development Team is decomposed by the end of this meeting, often to units of one day or less. The Development Team self-organizes to undertake the work in the Sprint Backlog, both during Sprint Planning and as needed throughout the Sprint.

The Product Owner can help to clarify the selected Product Backlog items and make trade-offs. If the Development Team determines it has too much or too little work, it may renegotiate the selected Product Backlog items with the Product Owner. The Development Team may also invite other people to attend in order to provide technical or domain advice.

By the end of the Sprint Planning, the Development Team should be able to explain to the Product Owner and Scrum Master how it intends to work as a self-organizing team to accomplish the Sprint Goal and create the anticipated Increment.

### 4.3. Sprint Goal

The Sprint Goal is an objective set for the Sprint that can be met through the implementation of Product Backlog. It provides guidance to the Development Team on why it is building the Increment. It is created during the Sprint Planning meeting. The Sprint Goal gives the Development Team some flexibility regarding the functionality implemented within the Sprint. The selected Product Backlog items deliver one coherent function, which can be the Sprint Goal. The Sprint Goal can be any other coherence that causes the Development Team to work together rather than on separate initiatives.

As the Development Team works, it keeps the Sprint Goal in mind. In order to satisfy the Sprint Goal, it implements the functionality and technology. If the work turns out to be different than the Development Team expected, they collaborate with the Product Owner to negotiate the scope of Sprint Backlog within the Sprint.

### 4.4. Daily Scrum

The Daily Scrum is a **15-minute** time-boxed event for the Development Team to synchronize activities and create a plan for the next 24 hours. This is done by inspecting the work since the last Daily Scrum and forecasting the work that could be done before the next one. The

Daily Scrum is held at the **same time and place each day** to reduce complexity.

During the meeting, the Development Team members explain:

- What did I do yesterday that helped the Development Team meet the Sprint Goal?

- What will I do today to help the Development Team meet the Sprint Goal?
- Do I see any impediment that prevents me or the Development Team from meeting the Sprint Goal?

The Development Team uses the Daily Scrum to inspect progress toward the Sprint Goal and to inspect how progress is trending toward completing the work in the Sprint Backlog. The Daily Scrum optimizes the probability that the Development Team will meet the Sprint Goal. Every day, the Development Team should understand how it intends to work together as a self-organizing team to accomplish the Sprint Goal and create the anticipated Increment by the end of the Sprint. The Development Team or team members often meet immediately after the Daily Scrum for detailed discussions, or to adapt, or replan, the rest of the Sprint's work.

The Scrum Master ensures that the Development Team has the meeting, but the Development Team is responsible for conducting the Daily Scrum. The Scrum Master teaches the Development Team to keep the Daily Scrum within the 15-minute time-box.

The Scrum Master enforces the rule that only Development Team members participate in the Daily Scrum.

Daily Scrums improve communications, eliminate other meetings, identify impediments to development for removal, highlight and promote quick decision-making, and improve the Development Team's level of knowledge. This is a key inspect and adapt meeting.

## 4.5. Sprint Review

A Sprint Review is held at the end of the Sprint to inspect the Increment and adapt the Product Backlog if needed. During the Sprint Review, the Scrum Team and stakeholders collaborate about what was done in the Sprint. Based on that and any changes to the Product Backlog during the Sprint, attendees collaborate on the next things that could be done to optimize value. This is an informal meeting, not a status meeting, and the presentation of the Increment is intended to elicit feedback and foster collaboration.

This is a **four-hour** time-boxed meeting for one-month Sprints. For shorter Sprints, the event is usually shorter. The Scrum Master ensures that the event takes place and that attendants understand its purpose. The Scrum Master teaches all to keep it within the time-box.

The Sprint Review includes the following elements:

- Attendees include the **Scrum Team** and **key stakeholders** invited by the Product Owner;
- The Product Owner explains what Product Backlog items have been "Done" and what has not been "Done";
- The Development Team discusses what went well during the Sprint, what problems it ran into, and how those problems were solved;
- The Development Team demonstrates the work that it has "Done" and answers questions about the Increment;
- The Product Owner discusses the Product Backlog as it stands. He or she projects likely completion dates based on progress to date (if needed);

- The entire group collaborates on what to do next, so that the Sprint Review provides valuable input to subsequent Sprint Planning;
- Review of how the marketplace or potential use of the product might have changed what is the most valuable thing to do next; and,
- Review of the timeline, budget, potential capabilities, and marketplace for the next anticipated release of the product.

The result of the Sprint Review is a revised Product Backlog that defines the probable Product Backlog items for the next Sprint. The Product Backlog may also be adjusted overall to meet new opportunities.

## 4.6. Sprint Retrospective

The Sprint Retrospective is an opportunity for the Scrum Team to inspect itself and create a plan for improvements to be enacted during the next Sprint.

The Sprint Retrospective occurs after the Sprint Review and prior to the next Sprint Planning. This is a three-hour time-boxed meeting for one-month Sprints. For shorter Sprints, the event is usually shorter. The Scrum Master ensures that the event takes place and that attendants understand its purpose. The Scrum Master teaches all to keep it within the time-box. The Scrum Master participates as a peer team member in the meeting from the accountability over the Scrum process.

The purpose of the Sprint Retrospective is to:

- **Inspect** how the **last Sprint** went with regards to people, relationships, process, and tools;
- **Identify** and order the major items that went well and potential **improvements**; and,
- Create a **plan** for implementing **improvements** to the way the Scrum Team does its work.

The Scrum Master encourages the Scrum Team to improve, within the Scrum process framework, its development process and practices to make it more effective and enjoyable for the next Sprint. During each Sprint Retrospective, the Scrum Team plans ways to increase product quality by adapting the definition of “Done” as appropriate.

By the end of the Sprint Retrospective, the Scrum Team should have identified improvements that it will implement in the next Sprint. Implementing these improvements in the next Sprint is the adaptation to the inspection of the Scrum Team itself. Although improvements may be implemented at any time, the Sprint Retrospective provides a formal opportunity to focus on inspection and adaptation.

## 5. Scrum Artifacts

Scrum's artifacts represent work or value to provide transparency and opportunities for inspection and adaptation. Artifacts defined by Scrum are specifically designed to maximize transparency of key information so that everybody has the same understanding of the artifact.

### 5.1. Product Backlog

The Product Backlog is an ordered list of everything that might be needed in the product and is the single source of requirements for any changes to be made to the product. The Product Owner is responsible for the Product Backlog, including its content, availability, and ordering.

A Product Backlog is never complete. The earliest development of it only lays out the initially known and best-understood requirements. The Product Backlog evolves as the product and the environment in which it will be used evolves. The Product Backlog is dynamic; it constantly changes to identify what the product needs to be appropriate, competitive, and useful. As long as a product exists, its Product Backlog also exists.

The Product Backlog lists all features, functions, requirements, enhancements, and fixes that constitute the changes to be made to the product in future releases. Product Backlog items have the attributes of a description, order, estimate and value.

As a product is used and gains value, and the marketplace provides feedback, the Product Backlog becomes a larger and more exhaustive list. Requirements never stop changing, so a Product Backlog is a living artifact. Changes in business requirements, market conditions, or technology may cause changes in the Product Backlog.

Multiple Scrum Teams often work together on the same product. One Product Backlog is used to describe the upcoming work on the product. A Product Backlog attribute that groups items may then be employed.

Product Backlog refinement is the act of adding detail, estimates, and order to items in the Product Backlog. This is an ongoing process in which the Product Owner and the Development Team collaborate on the details of Product Backlog items. During Product Backlog refinement, items are reviewed and revised. The Scrum Team decides how and when refinement is done. Refinement usually consumes no more than 10% of the capacity of the Development Team. However, Product Backlog items can be updated at any time by the Product Owner or at the Product Owner's discretion.

Higher ordered Product Backlog items are usually clearer and more detailed than lower ordered ones. More precise estimates are made based on the greater clarity and increased detail; the lower the order, the less detail. Product Backlog items that will occupy the Development Team for the upcoming Sprint are refined so that any one item can reasonably be "Done" within the Sprint time-box. Product Backlog items that can be "Done" by the Development Team within one Sprint are deemed "Ready" for selection in a Sprint Planning. Product Backlog items usually acquire this degree of transparency through the above described refining activities.

The Development Team is responsible for all estimates. The Product Owner may influence the Development Team by helping it understand and select trade-offs, but the people who will perform the work make the final estimate.

#### **5.1.1. Monitoring Progress Toward a Goal**

At any point in time, the total work remaining to reach a goal can be summed. The Product Owner tracks this total work remaining at least every Sprint Review. The Product Owner compares this amount with work remaining at previous Sprint Reviews to assess progress toward completing projected work by the desired time for the goal. This information is made transparent to all stakeholders.

Various projective practices upon trending have been used to forecast progress, like burn-downs, burn-ups, or cumulative flows. These have proven useful. However, these do not replace the importance of empiricism. In complex environments, what will happen is unknown. Only what has happened may be used for forward-looking decision-making.

### **5.2. Sprint Backlog**

The Sprint Backlog is the set of Product Backlog items selected for the Sprint, plus a plan for delivering the product Increment and realizing the Sprint Goal. The Sprint Backlog is a forecast by the Development Team about what functionality will be in the next Increment and the work needed to deliver that functionality into a “Done” Increment.

The Sprint Backlog makes visible all of the work that the Development Team identifies as necessary to meet the Sprint Goal.

The Sprint Backlog is a plan with enough detail that changes in progress can be understood in the Daily Scrum. The Development Team modifies the Sprint Backlog throughout the Sprint, and the Sprint Backlog emerges during the Sprint. This emergence occurs as the Development Team works through the plan and learns more about the work needed to achieve the Sprint Goal.

As new work is required, the Development Team adds it to the Sprint Backlog. As work is performed or completed, the estimated remaining work is updated. When elements of the plan are deemed unnecessary, they are removed. Only the Development Team can change its Sprint Backlog during a Sprint. The Sprint Backlog is a highly visible, real-time picture of the work that the Development Team plans to accomplish during the Sprint, and it belongs solely to the Development Team.

#### **5.2.1. Monitoring Sprint Progress**

At any point in time in a Sprint, the total work remaining in the Sprint Backlog can be summed. The Development Team tracks this total work remaining at least for every Daily Scrum to project the likelihood of achieving the Sprint Goal. By tracking the remaining work throughout the Sprint, the Development Team can manage its progress.

### **5.3. Increment**

The Increment is the sum of all the Product Backlog items completed during a Sprint and the value of the increments of all previous Sprints. At the end of a Sprint, the new Increment must be “Done,” which means it must be in useable condition and meet the Scrum Team’s definition of “Done.” It must be in useable condition regardless of whether the Product Owner decides to actually release it.

## 6. Artifact Transparency

Scrum relies on transparency. Decisions to optimize value and control risk are made based on the perceived state of the artifacts. To the extent that transparency is complete, these decisions have a sound basis. To the extent that the artifacts are incompletely transparent, these decisions can be flawed, value may diminish and risk may increase.

The Scrum Master must work with the Product Owner, Development Team, and other involved parties to understand if the artifacts are completely transparent. There are practices for coping with incomplete transparency; the Scrum Master must help everyone apply the most appropriate practices in the absence of complete transparency. A Scrum Master can detect incomplete transparency by inspecting the artifacts, sensing patterns, listening closely to what is being said, and detecting differences between expected and real results.

The Scrum Master's job is to work with the Scrum Team and the organization to increase the transparency of the artifacts. This work usually involves learning, convincing, and change. Transparency doesn't occur overnight, but is a path.



## 7. Definition of “Done”

When a Product Backlog item or an Increment is described as “Done”, everyone must understand what “Done” means. Although this varies significantly per Scrum Team, members must have a shared understanding of what it means for work to be complete, to ensure transparency. This is the definition of “Done” for the Scrum Team and is used to assess when work is complete on the product Increment.

The same definition guides the Development Team in knowing how many Product Backlog items it can select during a Sprint Planning. The purpose of each Sprint is to deliver Increments of potentially releasable functionality that adhere to the Scrum Team’s current definition of “Done.” Development Teams deliver an Increment of product functionality every Sprint. This Increment is useable, so a Product Owner may choose to immediately release it. If the definition of "done" for an increment **is** part of the conventions, standards or guidelines of the development organization, all Scrum Teams must follow it as a minimum. If "done" for an increment is **not** a convention of the development organization, the Development Team of the Scrum Team must define a definition of “done” appropriate for the product. If there are multiple Scrum Teams working on the system or product release, the **development teams** on all of the Scrum Teams **must mutually define** the definition of “Done.”

Each Increment is additive to all prior Increments and thoroughly tested, ensuring that all Increments work together.

As Scrum Teams mature, it is expected that their definitions of “Done” will expand to include more stringent criteria for higher quality. Any one product or system should have a definition of “Done” that is a standard for any work done on it.

## 8. Q and A

1. Who is responsible for managing the progress of work during a Sprint?
  - A) The Development Team
  - B) The Scrum Master
  - C) The Product Owner
  - D) The most junior member of the Team
2. The maximum length of the Sprint Review (its time-box) is:
  - A) 2 hours.
  - B) 4 hours for a monthly Sprint. For shorter Sprints it is usually shorter.
  - C) As long as needed.
  - D) 1 day.
  - E) 4 hours and longer as needed.
3. The length of a Sprint should be:
  - A) Short enough to keep the business risk acceptable to the Product Owner.
  - B) Short enough to be able to synchronize the development work with other business events.
  - C) No more than one calendar month.
  - D) All of these answers are correct.
4. Scrum does not have a role called "project manager."
  - A) True
  - B) False
5. When might a Sprint be abnormally terminated?
  - A) When it becomes clear that not everything will be finished by the end of the Sprint.
  - B) When the Development Team feels that the work is too hard.
  - C) When the sales department has an important new opportunity.
  - D) When the Sprint Goal becomes obsolete.
6. The Product Backlog is ordered by:
  - A) Size, where small items are at the top and large items are at the bottom.
  - B) Risk, where safer items are at the top, and riskier items are at the bottom
  - C) Least valuable items at the top to most valuable at the bottom.
  - D) Items are randomly arranged.
  - E) Whatever is deemed most appropriate by the Product Owner.
7. Who should know the most about the progress toward a business objective or a release, and be able to explain the alternatives most clearly?
  - A) The Product Owner
  - B) The Development Team

- C) The Scrum Master
- D) The Project Manager

8. An organization has decided to adopt Scrum, but management wants to change the terminology to fit with terminology already used. What will likely happen if this is done?
- A) Without a new vocabulary as a reminder of the change, very little change may actually happen.
  - B) The organization may not understand what has changed with Scrum and the benefits of Scrum may be lost.
  - C) Management may feel less anxious.
  - D) All answers apply.
9. The CEO asks the Development Team to add a "very important" item to a Sprint that is in progress. What should the Development Team do?
- A) Add the item to the current Sprint without any adjustments.
  - B) Add the item to the current Sprint and drop an item of equal size.
  - C) Add the item to the next Sprint.
  - D) Inform the Product Owner so he/she can work with the CEO.
10. The time-box for a Daily Scrum is?
- A) The same time of day every day.
  - B) Two minutes per person.
  - C) 4 hours.
  - D) 15 minutes.
  - E) 15 minutes for a 4 week sprint. For shorter Sprints it is usually shorter.
11. When does the next Sprint begin?
- A) Next Monday.
  - B) Immediately following the next Sprint Planning.
  - C) When the Product Owner is ready.
  - D) Immediately after the conclusion of the previous Sprint.
12. What are the two primary ways a Scrum Master keeps a Development Team working at its highest level of productivity?
- A) By facilitating Development Team decisions
  - B) By removing impediments that hinder the Development Team
  - C) By ensuring the meetings start and end at the proper time
  - D) By keeping high value features high in the Product Backlog
13. When multiple teams work together on the same product, each team should maintain a separate Product Backlog.
- A) True
  - B) False
14. The purpose of a Sprint is to produce a done increment of working product.

- A) True
- B) False

**15.** Upon what type of process control is Scrum based?

- A) Empirical
- B) Hybrid
- C) Defined
- D) Complex

**16.** Which statement best describes a Product Owner's responsibility?

- A) Optimizing the value of the work the Development Team does.
- B) Directing the Development Team.
- C) Managing the project and ensuring that the work meets the commitments to the stakeholders.
- D) Keeping stakeholders at bay.

**17.** Why is the Daily Scrum held at the same time and same place?

- A) The place can be named.
- B) The consistency reduces complexity.
- C) The Product Owner demands it.
- D) Rooms are hard to book and this lets it be booked in advance.

**18.** The time-box for the Sprint Planning meeting is?

- A) 4 hours.
- B) 8 hours for a monthly Sprint. For shorter Sprints it is usually shorter.
- C) Whenever it is done.
- D) Monthly.

**19.** Which of the below are roles on a Scrum Team?

- A) Development Team
- B) Users
- C) Customers
- D) Product Owner
- E) Scrum Master

**20.** Development Team membership should change:

- A) Every Sprint to promote shared learning.
- B) Never, because it reduces productivity.
- C) As needed, while taking into account a short term reduction in productivity.
- D) As needed, with no special allowance for changes in productivity.

**21.** What is the main reason for the Scrum Master to be at the Daily Scrum?

- A) To make sure every team member answers the three questions.

- B) He or she does not have to be there; he or she only has to ensure the Development Team has a Daily Scrum.
- C) To write down any changes to the Sprint Backlog, including adding new items, and tracking progress on the burn-down.
- D) To gather status and progress information to report to management.

**22.** Who is required to attend the Daily Scrum?

- A) The Development Team.
- B) The Scrum team.
- C) The Development Team and Scrum Master.
- D) The Development Team and Product Owner.
- E) The Scrum Master and Product Owner.

**23.** Who has the final say on the order of the Product Backlog?

- A) The Stakeholders
- B) The Development Team
- C) The Scrum Master
- D) The Product Owner
- E) The CEO

**24.** It is mandatory that the product increment be released to production at the end of each Sprint.

- A) True
- B) False

**25.** When many Development Teams are working on a single product, what best describes the definition of "done?"

- A) Each Development Team defines and uses its own. The differences are discussed and reconciled during a hardening Sprint.
- B) Each Development Team uses its own but must make their definition clear to all other Teams so the differences are known.
- C) All Development Teams must have a definition of "done" that makes their combined work potentially releasable.
- D) It depends.

**26.** When is a Sprint over?

- A) When all Product Backlog items meet their definition of done.
- B) When the Product Owner says it is done.
- C) When all the tasks are completed.
- D) When the time-box expires.

**27.** During a Sprint, a Development Team determines that it will not be able to finish the complete forecast. Who should be present to review and adjust the Sprint work selected?

- A) The Scrum Master, the project manager and the Development Team.

- B) The Product Owner and the Development Team.
- C) The Product Owner and all stakeholders.
- D) The Development Team.

**28.** Who is on the Scrum Team?

- A) The Scrum Master
- B) The Product Owner
- C) The Development Team
- D) Project Manager
- E) None of the above

**29.** Development Team members volunteer to own a Sprint Backlog item:

- A) At the Sprint planning meeting.
- B) Never. All Sprint Backlog Items are "owned" by the entire Development Team, even though each one may be done by an individual development team member.
- C) Whenever a team member can accommodate more work.
- D) During the Daily Scrum.

**30.** The Development Team should have all the skills needed to:

- A) Complete the project as estimated when the date and cost are committed to the Product Owner.
- B) Do all of the development work, except for specialized testing that requires additional tools and environments.
- C) Turn the Product Backlog items it selects into an increment of potentially shippable product functionality.

**31.** Which two (2) things does the Development Team do during the first Sprint?

- A) Deliver an increment of releasable software.
- B) Determine the complete architecture and infrastructure for the product.
- C) Develop and deliver at least one piece of functionality.
- D) Develop a plan for the rest of the release.
- E) Create the complete Product Backlog to be developed in subsequent Sprints.

**32.** What is the recommended size for a Development Team (within the Scrum Team)?

- A) Minimal 7
- B) 3 to 9
- C) 7 plus or minus 2
- D) 9

**33.** What is the role of Management in Scrum?

- A) Continually monitor staffing levels of the Development Team.
- B) Monitor the Development Team's productivity.
- C) Support the Product Owner with insights and information into high value product and system capabilities. Support the Scrum Master to cause organizational change that

fosters empiricism, self-organization, bottom-up intelligence, and intelligent release of software.

D) Identify and remove people that aren't working hard enough.

**34.** The Development Team should not be interrupted during the Sprint. The Sprint Goal should remain intact. These are conditions that foster creativity, quality and productivity. Based on this, which of the following is **FALSE**?

A) The Product Owner can help clarify or optimize the Sprint when asked by the Development Team.

B) The Sprint Backlog is fully formulated in the Sprint Planning meeting and does not change during the Sprint.

C) As a decomposition of the selected Product Backlog Items, the Sprint Backlog changes and may grow as the work emerges.

D) The Development Team may work with the Product Owner to remove or add work if it finds it has more or less capacity than it expected.

**35.** How much work must a Development Team do to a Product Backlog item it selects for a Sprint?

A) As much as it has told the Product Owner will be done for every Product Backlog item it selects in conformance with the definition of done.

B) As much as it can fit into the Sprint.

C) All development work and at least some testing.

D) Analysis, design, programming, testing and documentation.

**36.** Which statement best describes the Sprint Review?

A) It is a mechanism to control the Development Team's activities during a Sprint.

B) It is when the Scrum Team and stakeholders inspect the outcome of a Sprint and figure out what to do next.

C) It is a demo at the end of the Sprint for everyone in the organization to check on the work done.

**37.** Which statement best describes Scrum?

A) A complete methodology that defines how to develop software.

B) A cookbook that defines best practices for software development.

C) A framework within which complex products in complex environments are developed.

D) A defined and predictive process that conforms to the principles of Scientific Management.

**38.** What does it mean to say that an event has a time-box?

A) The event must happen at a set time.

B) The event must happen by a given time.

C) The event must take at least a minimum amount of time

D) The event can take no more than a maximum amount of time.

**39.** The three pillars of empirical process control are:

- A) Respect For People, Kaizen, Eliminating Waste
- B) Planning, Demonstration, Retrospective
- C) Inspection, Transparency, Adaptation
- D) Planning, Inspection, Adaptation
- E) Transparency, Eliminating Waste, Kaizen

**40.** During the Daily Scrum, the Scrum Master's role is to:

- A) Lead the discussions of the Development Team.
- B) Make sure that all 3 questions have been answered.
- C) Manage the meeting in a way that each team member has a chance to speak.
- D) Teach the Development Team to keep the Daily Scrum within the 15 minute time-box.
- E) All answers apply.



Questions	Answers	Comments
1	A	The Development Team uses the Daily Scrum to inspect progress toward the Sprint Goal and to inspect how progress is trending toward completing the work in the Sprint Backlog.
2	B	Sprint Review is a four-hour time-boxed meeting for one-month Sprints. For shorter Sprints, the event is usually shorter.
3	D	All of these choices are appropriate considerations in determining the length of a Sprint.
4	A	A Scrum Team has a Scrum Master, a Product Owner and a Development Team. As a whole they have all controls needed.
5	D	A Sprint can be cancelled before the Sprint time-box is over. A Sprint would be cancelled if the Sprint Goal becomes obsolete. This might occur if the company changes direction or if market or technology conditions change.
6	E	The Product Owner decides what makes the most sense to optimize the value of the work being done by the Development Team.
7	A	The Product Owner is the sole person responsible for managing the Product Backlog, which includes that the Product Backlog is visible, transparent, and clear to all, and shows what the Scrum Team will work on next.
8	D	
9	D	The items selected for a Sprint have been selected as most valuable with the Product Owner. The items serve the Sprint's goal. No changes should be made that endanger the Sprint Goal. No one external to the Scrum Team can force changes on the Development Team (Sprint Backlog) and the Product Owner (Product Backlog).
10	D	The length of a daily Scrum is fixed at 15 minutes, it does not change with the length of a Sprint.
11	D	A new Sprint starts immediately after the conclusion of the previous Sprint.
12	A, B	A Scrum Master is a servant-leader for the Development Team. Facilitation and removing impediments serves a team in achieving the best productivity possible.
13	B	Products have one Product Backlog, regardless of how many teams are used. Any other setup makes it difficult for the Development Team to determine what it should work on.
14	A	The heart of Scrum is a Sprint, a time-box of one month or less during which a "Done", usable, and potentially releasable product Increment is created.
15	A	Scrum is founded on empirical process control theory, or empiricism. Empiricism asserts that knowledge comes from experience and making decisions based on what is

		known.
16	A	The Product Owner is responsible for maximizing the value of the product and the work of the Development Team.
17	B	The Daily Scrum is held at the same time and place each day to reduce complexity.
18	B	Sprint Planning is time-boxed to a maximum of eight hours for a one-month Sprint. For shorter Sprints, the event is usually shorter.
19	A, D, E	The Scrum Team consists of a Product Owner, the Development Team, and a Scrum Master.
20	C	Teams typically go through some steps before achieving a state of increased performance. Changing membership typically reduces cohesion, affecting performance and productivity in the short term.
21	B	The Scrum Master enforces the rule that only Development Team members participate in the Daily Scrum.
22	A	Only the people doing the work described on the Sprint Backlog need to inspect and adapt at the Daily Scrum. If the Scrum Master or Product Owner is also on the Development Team, they will need to be at the Daily Scrum. Otherwise, the Scrum Master simply has to make sure the Development Team knows how to conduct a Daily Scrum and does so.
23	D	The Product Owner is the sole person responsible for managing the Product Backlog.
24	B	The product increment should be usable and releasable at the end of every Sprint, but it does not have to be released.
25	C	Scrum requires an Increment to be releasable. This is an Increment of product. Many teams working on a single product are expected to deliver such an Increment.
26	D	The duration of a Sprint is fixed and cannot be shortened or lengthened.
27	B	During the Sprint, scope may be clarified and re-negotiated between the Product Owner and Development Team as more is learned.
28	A, B, C	The Scrum Team consists of the Scrum Master (manages the process), the Product Owner (decides what to do) and the Development Team (does the work).
29	B	Sprint Backlog and all of its items are collectively owned by the Development Team. No individual team member can claim ownership over an item as this would block communication and collaboration.
30	C	The Development Team consists of professionals who do the work of delivering a potentially releasable Increment of "Done" product at the end of each Sprint. Development Teams are cross-functional, with all of the skills as a team necessary to create a product Increment.

31	A, C	The heart of Scrum is a Sprint, a time-box of one month or less during which a "Done", useable, and potentially releasable product Increment is created. This applies to every Sprint
32	B	Optimal Development Team size is small enough to remain nimble and large enough to complete significant work. Fewer than three Development Team members decreases interaction and results in smaller productivity gains. More than nine members simply requires too much coordination.
33	C	Management has no active role in the actual product development through Scrum. However, management external to the Scrum team is incredibly important in setting the vision and strategy to guide the overall direction of the organization.
34	B	The Sprint Backlog makes visible all of the work that the Development Team identifies as necessary to meet the Sprint Goal. The Development Team modifies the Sprint Backlog throughout the Sprint, and the Sprint Backlog emerges during the Sprint.
35	A	The purpose of each Sprint is to deliver Increments of potentially releasable functionality that adhere to the Scrum Team's current definition of "Done".
36	B	Every event in Scrum, besides the Sprint which is a container for the other events, is an opportunity to Inspect AND Adapt.
37	C	Scrum is not a process or a technique for building products; rather, it is a framework within which you can employ various processes and techniques.
38	D	Time-boxed events are events that have a maximum duration.
39	C	Scrum is founded on empirical process control theory, or empiricism. Empiricism asserts that knowledge comes from experience and making decisions based on what is known. Three pillars uphold every implementation of empirical process control: transparency, inspection, and adaptation.
40	D	The Scrum Master ensures that the Development Team has the meeting, but the Development Team is responsible for conducting the Daily Scrum. The Scrum Master teaches the Development Team to keep the Daily Scrum within the 15-minute time-box. The Scrum Master enforces the rule that only Development Team members participate in the Daily Scrum.

## 9. Glossary

**ALM (Application Lifecycle Management):** holistic view on the management of software applications and systems, accounting for all stages of the existence of a software product.

**ATDD (Acceptance Test-Driven Development):** test-first software development practice in which acceptance criteria for new functionality are created as automated tests. The failing tests are constructed to pass as development proceeds and acceptance criteria are met.

**BDD (Behavior-Driven Development):** agile software development practice adding to TDD the description of the desired functional behavior of the new functionality.

**Branching:** creating a logical or physical copy of code within a version control system so that this copy might be changed in isolation.

**Burn-down Chart:** a chart showing the evolution of remaining effort against time. Burn-down charts are an optional implementation within Scrum to make progress transparent.

**Burn-up Chart:** a chart showing the evolution of an increase in a measure against time. Burn-up charts are an optional implementation within Scrum to make progress transparent.

**Clean Code:** software code that is expressed well, formatted correctly, and organized for later coders to understand. Clarity is preferred over cleverness.

**Code Coverage:** a measurement indicating the amount of product code that is exercised by tests.

**Cohesion and Coupling:** coupling refers to the interdependencies between modules, while cohesion describes how related the functions within a single module are.

**Collective Code Ownership:** a software development principle popularized by Extreme Programming holding that all contributors to a given codebase are jointly responsible for the code in its entirety.

**Continuous Delivery:** a software delivery practice similar to Continuous Deployment except a human action is required to promote changes into a subsequent environment along the pipeline.

**Continuous Deployment:** a software delivery practice in which the release process is fully automated in order to have changes promoted to the production environment with no human intervention.<

**Continuous Integration (CI):** agile software development practice popularized by Extreme Programming in which newly checked-in code is built, integrated and tested frequently, generally multiple times a day.

**Cyclomatic Complexity:** a measure of code complexity based on the number of independent logical branches through a code base. Cyclomatic complexity is expressed as a simple integer.

**Cross-functional:** characteristic of a team holding that all the skills required to successfully produce a releasable Increment in a sprint are available within the team, where releasable refers to making the software available in production.

**Daily Scrum:** daily time-boxed event of 15 minutes, or less, for the Development Team to re-plan the next day of development work during a Sprint. Updates are reflected in the Sprint Backlog.

**Definition of Done:** a shared understanding of expectations that software must live up to in order to be releasable into production. Managed by the Development Team.

**Developer:** any member of a Development Team, regardless of technical, functional or other specialty.

**DevOps:** an organizational concept serving to bridge the gap between development and operations, in terms of skills, mind-set, practices and silo-mentality. The underlying idea is that developers are aware of, and in daily work consider implications on operations, and vice versa.

**Development Team:** the role within a Scrum Team accountable for managing, organizing and doing all development work required to create a releasable Increment of product every Sprint.

**DRY (don't repeat yourself):** software development principle to avoid repetition of the same information in one system, preventing the same code from being produced multiple times on a code base.

**Emergence:** the process of the coming into existence or prominence of new facts or new knowledge of a fact, or knowledge of a fact becoming visible unexpectedly.

**Empiricism:** process control type in which only the past is accepted as certain and in which decisions are based on observation, experience and experimentation. Empiricism has three pillars: transparency, inspection and adaptation.

**Engineering standards:** a shared set of development and technology standards that a Development Team applies to create releasable Increments of software.

**Extreme Programming (XP):** agile software development framework with an extreme focus on programming and taking engineering practices to an extreme in order to create and release high quality code. Highly complementary to the Scrum framework.

**Forecast (of functionality):** the selection of items from the Product Backlog a Development Team deems feasible for implementation in a Sprint.

**Feature Toggle:** software development practice that allows dynamically turning (parts of) functionality on and off without impacting the overall accessibility of the system by its users.

**Increment:** a piece of working software that adds to previously created Increments, where the sum of all Increments - as a whole - form a product.

**Pair Programming:** agile software development practice popularized by Extreme Programming in which 2 team members jointly create new functionality.

**Product Backlog:** an ordered list of the work to be done in order to create, maintain and sustain a product. Managed by the Product Owner.

**Product Backlog refinement:** the activity in a Sprint through which the Product Owner and the Development Team add granularity to the Product Backlog.

**Product Owner:** the role in Scrum accountable for maximizing the value of a product, primarily by incrementally managing and expressing business and functional expectations for a product to the Development Team(s).

**Ready:** a shared understanding by the Product Owner and the Development Team regarding the preferred level of description of Product Backlog items introduced at Sprint Planning.

**Refactoring:** agile software development practice popularized by Extreme Programming in which code is adjusted within the code base without impacting the external, functional behavior of that code.

**Scout Rule:** the practice of always leaving the code base in a little better state than it was found before modifications. A means to progress towards Clean Code.

**Scrum:** a framework to support teams in complex product development. Scrum consists of Scrum Teams and their associated roles, events, artifacts, and rules, as defined in the Scrum Guide™.

**Scrum Board:** a physical board to visualize information for and by the Scrum Team, often used to manage Sprint Backlog. Scrum boards are an optional implementation within Scrum to make information visible.

**Scrum Guide™:** the definition of Scrum, written and provided by Ken Schwaber and Jeff Sutherland, co-creators of Scrum. This definition consists of Scrum's roles, events, artifacts, and the rules that bind them together.

**Scrum Master:** the role within a Scrum Team accountable for guiding, coaching, teaching and assisting a Scrum Team and its environments in a proper understanding and use of Scrum.

**Scrum Team:** a self-organizing team consisting of a Product Owner, Development Team and Scrum Master.

**Scrum Values:** a set of fundamental values and qualities underpinning the Scrum framework; commitment, focus, openness, respect and courage.

**Self-organization:** the management principle that teams autonomously organize their work. Self-organization happens within boundaries and against given goals. Teams choose how best to accomplish their work, rather than being directed by others outside the team.

**Specification by Example:** agile software development practice based on TDD and ATDD that calls for using realistic examples from past experience instead of untested or abstract statements in the description of the desired functional behavior.

**Sprint:** time-boxed event of 30 days, or less, that serves as a container for the other Scrum events and activities. Sprints are done consecutively, without intermediate gaps.

**Sprint Backlog:** an overview of the development work to realize a Sprint's goal, typically a forecast of functionality and the work needed to deliver that functionality. (Managed by the Development Team)

**Sprint Goal:** a short expression of the purpose of a Sprint, often a business problem that is addressed. Functionality might be adjusted during the Sprint in order to achieve the Sprint Goal.

**Sprint Planning:** time-boxed event of 1 day, or less, to start a Sprint. It serves for the Scrum Team to inspect the work from the Product Backlog that's most valuable to be done next and design that work into Sprint backlog.

**Sprint Retrospective:** time-boxed event of 3 hours, or less, to end a Sprint. It serves for the Scrum Team to inspect the past Sprint and plan for improvements to be enacted during the next Sprint.

**Sprint Review:** time-boxed event of 4 hours, or less, to conclude the development work of a Sprint. It serves for the Scrum Team and the stakeholders to inspect the Increment of product resulting from the Sprint, assess the impact of the work performed on overall progress and update the Product backlog in order to maximize the value of the next period.

**Stakeholder:** a person external to the Scrum Team with a specific interest in and knowledge of a product that is required for incremental discovery. Represented by the Product Owner and actively engaged with the Scrum Team at Sprint Review.

**TDD (Test-Driven Development):** test-first software development practice in which test cases are defined and created first, and subsequently executable code is created to make the test pass. The failing tests are constructed to pass as development proceeds and tests succeed.

**Technical Debt:** the typically unpredictable cost of having to resolve or work with unfinished, unclear, undone or defect code once it has been released into production.

**User Story:** agile software development practice from Extreme Programming to express requirements from an end user perspective, emphasising verbal communication. In Scrum, it is often used to express functional items on the Product Backlog.

**Unit Test:** low-level technical test focusing on small parts of a software system that can be executed fast and in isolation. The definition and boundaries of a 'unit' generally depends on the context and is to be agreed by the Development Team.

**Velocity:** an optional, but often used, indication of the average amount of Product Backlog turned into an Increment of product during a Sprint by a Scrum Team, tracked by the Development Team for use within the Scrum Team.

*Also read "Scrum - A Pocket Guide (Best Practice)" by Gunter Verheyen.*

**End of document**