

Scrum Alliance®

Certified ScrumMaster®

Learning Objectives

February 2017

by the Scrum Alliance CSM® and CSP® Learning Objectives Committees

Introduction

Purpose

This document describes the Learning Objectives (LOs) that must be covered in a Certified ScrumMaster (CSM) course. These Learning Objectives take the following into consideration:

- Every implementation of Scrum is different.
- Teams and organizations apply Scrum within their context, but the fundamental framework always remains the same.

The Learning Objectives for this course are based on:

- *Scrum Guide*, <http://scrumguides.org>
- Agile Manifesto, 4 values and 12 principles, <http://www.agilemanifesto.org>
- Scrum values, <https://www.scrumalliance.org/why-scrum/core-scrum-values-roles>

Scope

Scrum Alliance has adopted the *Scrum Guide, The Definitive Guide to Scrum: The Rules of the Game*, coauthored and updated (most recently in 2016) by the co-creators of the Scrum framework, as the guiding curriculum for this course. CSM and CSPO® candidates are expected to build a body of knowledge of the Scrum framework, including its roles, events, and artifacts. Incorporating Scrum principles and practices takes diligence, patience, and a commitment to continuous improvement. Scrum is a framework, not a prescriptive methodology.

Students in a CSM course should expect that each Learning Objective identified in this document will be covered in a CSM course. The CSM Learning Objectives fall into the following categories:

1. **Lean, Agile, and Scrum**
2. **Agile Facilitation**
3. **Agile Coaching**
4. **Service to the Development Team**
5. **Service to the Product Owner**
6. **Service to the Organization**

Individual trainers (CSTs) or coaches (CECs) may choose to teach ancillary topics. Ancillary topics presented in a CSM course must be clearly indicated as such.

Learning Objectives

A note about examples used in the following Learning Objectives:

Several Learning Objectives include a list of examples. The examples are used to clarify the intent of the objective. Individual trainers or coaches can use the provided examples, their own examples that still meet the objective, or a mix of both. Examples do not imply that they are the only options, nor that they constitute an exhaustive list.

A note about Bloom's Taxonomy:

While some Learning Objectives appear to tell the trainer how to teach, that is not the intent. Bloom's-style Learning Objectives describe what the learner can do upon completing the class. Rather than include that text in each Learning Objective, please mentally append the following phrase to each objective:

"Upon successful completion of the CSM course, the learner will be able to ..."

1. Lean, Agile, and Scrum

Values and Principles

- 1.1. ... describe Scrum's relationship to the Agile Manifesto.

Scrum Theory

- 1.2. ... define empirical process control and list the three pillars.
- 1.3. ... describe how the values of Scrum — focus, courage, commitment, openness, and respect — are present in a specific Scrum event, artifact, or role.
- 1.4. ... explain why Scrum is framework and list two ways that a framework is different from a process\methodology.
- 1.5. ... explain how evolutionary product planning in an empirical environment differs from traditional fixed planning, and give an example of when each may be appropriate.

The Scrum Team

- 1.6. ... illustrate the three roles in a Scrum Team and how they interact with each other to deliver the product increment within a sprint.
- 1.7. ... describe at least three disadvantages of shared roles in Scrum (e.g., a Development Team member being the Scrum Master or Product Owner, or the Product Owner being the Scrum Master).

The Product Owner (PO)

- 1.8. ... list at least three prerogatives (e.g., independent authority, defining scope for the Development Team, deciding when to release) and five responsibilities (e.g., ongoing visioning, ordering, budgeting, and dates; maximizing the value of the Development Team's work; maximizing the value of the Scrum Team's product) of the Product Owner.
- 1.9. ... discuss at least two reasons why the Product Owner is a single person and not a group or a committee.
- 1.10. ... discuss how and why the Product Owner maintains authority over the product (i.e., contents and order of the product backlog, when to release and when to stop developing) while working collaboratively with the Development Team and stakeholders to gather their ideas, feedback, and input.

The Development Team (Dev Team)

- 1.11. ... list at least three prerogatives (e.g., produce quality work, provide their own estimates, sign up for work rather than be assigned work) and five responsibilities (e.g., produce a "Done," usable, releasable product increment each and every sprint, improve their engineering practices, estimate, assist the Product Owner in maintaining the product backlog, implement action items) of the Development Team.
- 1.12. ... list at least five characteristics of the Development Team: self-organizing, cross-functional, no titles other than Developer, no subteams, and mutual accountability.
- 1.13. ... identify at least three negative consequences that arise when the Development Team consists of fewer than three or more than nine people.

- 1.14. ... identify at least two reasons why only the Product Owner can offer work to the Development Team (e.g., interrupts the Development Team's focus, undermines the Product Owner's independent authority).

The Scrum Master

- 1.15. ... list at three prerogatives (e.g., experiment with new ideas, have access to stakeholders and decision makers, address issues openly) and five responsibilities (e.g., ensure that Scrum is understood and enacted within the organization, act as a change agent, coach the Scrum Team and the organization, increase visibility, lead through influence) of the Scrum Master.
- 1.16. ... identify at least three reasons why the Scrum Master has no authority but leads through influence.

Scrum Events and Artifacts

- 1.17. ... list at least three benefits to timeboxing.
- 1.18. ... give one example of how a Scrum Team will inspect and adapt and increase transparency at each of the Scrum events.

Sprint and Increment

- 1.19. ... discuss a scenario when a Product Owner may consider sprint cancellation and identify at least two alternatives.
- 1.20. ... describe at least two reasons why the scope and duration of a sprint are fixed (e.g., promotes the successful delivery of the sprint goal, supports the Scrum Team to learn how to deliver valuable increments iteratively).
- 1.21. ... define the outcome of every sprint (e.g., a potentially releasable product increment that adheres to the current Definition of Done) and describe at least three reasons why that is important.
- 1.22. ... discuss at least three reasons why the increment must be brought to the current Definition of Done regardless of whether the Product Owner chooses to release the increment.

Sprint Planning

- 1.23. ... list the participants, one input (i.e., "ready" product backlog items), and at least two outputs (i.e., sprint goal, sprint backlog) of sprint planning.
- 1.24. ... describe at least three responsibilities for the Development Team (i.e., decide how much work is brought into the sprint, offer a forecast, create sprint backlog items), Product Owner (i.e., offer the sprint goal, provide clarification, negotiate scope), and Scrum Master (i.e., facilitate the dialogue between the team and the Product Owner, maintain the timebox, ensure that the Development Team neither overcommits nor undercommits) during sprint planning.
- 1.25. ... discuss the focus of the activities of the Product Owner and Development Team during the two topics of sprint planning: the "What" and the "How."
- 1.26. ... give an example of a sprint goal and identify at least two benefits of having a sprint goal (e.g., provides greater context for the work, helps stakeholders understand why they are being asked to participate in a sprint review).
- 1.27. ... discuss at least three negative impacts that arise when the Scrum Team disregards one or more of the elements of sprint planning.

Daily Scrum

- 1.28. ... discuss at least three ways the Daily Scrum differs from a traditional status meeting and why the various constraints (e.g., no more than 15 minutes, meets every day, only the Development Team members participate) exist to support the Scrum Team.
- 1.29. ... describe at least three responsibilities for the Development Team (e.g., answer the three questions, review their progress toward the sprint goal, update the sprint backlog) during the Daily Scrum, and describe contributions that may be made by the Product Owner (e.g., provide clarification, offer early feedback, give the team the freedom to organize their own work) and Scrum Master (e.g., teach the Development Team how to run the Daily Scrum; offer observations, not solutions; facilitate the conversation if necessary).
- 1.30. ... list the three questions associated with the Daily Scrum agenda and identify one reason why the Development Team's responses in the Daily Scrum are linked to the sprint goal.
- 1.31. ... discuss at least three negative impacts that arise when the Scrum Team disregards one or more of the elements of the Daily Scrum.

Product Backlog Refinement

- 1.32. ... discuss at least two reasons why the Development Team spends no more than 10% of their capacity on product backlog refinement.
- 1.33. ... discuss at least three negative impacts that arise when the Scrum Team disregards one or more of the elements of product backlog refinement.

Sprint Review

- 1.34. ... list the participants of the sprint review and describe at least two responsibilities for the Development Team (e.g., demonstrate the results of the sprint, respond to questions), Product Owner (e.g., explain which product backlog items have been "Done" and which have not, track total work remaining toward a goal), Scrum Master (ensure that the event takes place, maintain the timebox), and stakeholders (e.g., provide feedback, help resolve impediments) during the sprint review.
- 1.35. ... explain at least four of the review activities that take place during the sprint review that pertain to work beyond what has been completed in the sprint, for example: time line, budget, potential use in the marketplace, product backlog, release schedule.
- 1.36. ... discuss at least three things that do not occur at a sprint review (e.g., formal sign-off, presentation of product backlog items that do not meet the Definition of Done, discussion of work hours, a mere demonstration of new functionality with feedback).
- 1.37. ... identify at least three outcomes for a sprint review (e.g., revised product backlog, release the increment, cancel further development).
- 1.38. ... discuss at least three negative impacts that arise when the Scrum Team disregards one or more of the elements of the sprint review.

Sprint Retrospective

- 1.39. ... list the participants of the sprint retrospective and describe at least two responsibilities for the Development Team (e.g., review the results of the previous sprint, look for ways to improve their development practices), Product Owner (e.g., offer observations on the results of the previous sprint,

- participate in retrospective discussions), and Scrum Master (ensure that the event takes place, maintain the timebox) during the sprint retrospective.
- 1.40. ... discuss at least three negative impacts that arise when the Scrum Team disregards one or more of the elements of the sprint retrospective.

Product Backlog

- 1.41. ... describe at least two responsibilities of the Development Team (e.g., provide estimates), Product Owner (e.g., provide clarification), and Scrum Master (e.g., demonstrate useful item formulation methods) in the development and maintenance of the product backlog.
- 1.42. ... identify at least three essential characteristics of the product backlog (e.g., dynamic, ordered, having varying levels of detail).
- 1.43. ... list at least four elements of a product backlog Item (e.g., description, order, estimate, value).

Sprint Backlog

- 1.44. ... identify at least three essential characteristics of the sprint backlog (e.g., just enough detail, highly visible, real-time snapshot of the Development Team's work for the sprint).
- 1.45. ... explain how the sprint backlog can be changed, how responsibilities for making changes are divided among the Scrum Team members, and the limits of these changes (e.g., the Development Team can add/remove/modify items at any time, the Product Owner can remove selected product backlog items and thus modify the sprint backlog indirectly, the Scrum Master can make suggestions about missing items to fulfill the Definition of Done).

Definition of Done

- 1.46. ... identify at least two reasons why multiple teams working on the same product backlog must have a shared and consistent Definition of Done.
- 1.47. ... discuss why the Definition of Done evolves over time and identify at least three places in Scrum where the Scrum Team might note weaknesses in their Definition of Done.
- 1.48. ... explain the importance of a strong Definition of Done and describe at least two risks associated with a weaker Definition of Done.
- 1.49. ... facilitate one way to create a Definition of Done.

2. Agile Facilitation

- 2.1. ... list at least three ways that the Scrum Master facilitates for the Scrum Team.
- 2.2. ... demonstrate at least three techniques for facilitating group decision making (dot voting, fist of five, thumb voting).
- 2.3. ... describe, using two concrete examples, when the Scrum Master should *not* act as the facilitator for the Scrum Team.
- 2.4. ... describe three obstacles to clear communication and describe their impacts on both the sender and receiver (e.g., sarcasm, irony, aggressiveness, defensiveness, misdirection).

- 2.5. ... describe at least four ground rules to foster clear communication in a collaborative meeting and describe how the introduction of the ground rules impacts the interaction.

3. Agile Coaching

- 3.1. ... repeat the difference between facilitating, teaching, mentoring, and coaching.
- 3.2. ... list at least three different challenges facing a self-organizing team (e.g., bad forecast, technical debt, someone is leaving the team).
- 3.3. ... practice one approach used in a retrospective that could help resolve a challenge faced by a self-organizing team.

4. Service to the Development Team

Scrum Master as Servant-Leader

- 4.1. ... define Servant Leadership and discuss at least three ways in which it is different from authoritarian, top-down management.
- 4.2. ... describe three scenarios where the Scrum Master acts as the Servant-Leader for the Scrum Team and/or organization.
- 4.3. ... discuss a scenario in which the Scrum Master, acting as a Servant-Leader, improved one aspect of the Scrum Team.

Value of Engineering Practices

- 4.4. ... define technical debt and explain the impact of accumulating technical debt (e.g., by showing how technical debt impacts the capacity of the team over time, the increase of cost in addressing technical debt too late, using the “Debt Quadrant” by Martin Fowler).
- 4.5. ... list at least five practices (e.g., from Extreme Programming: test-driven development, pair programming, continuous integration, collective code ownership, refactoring) that will help Scrum Teams deliver a high-quality product increment and reduce technical debt each sprint.
- 4.6. ... list at least three ways technical practices may impact the Development Team’s ability to deliver a potentially releasable increment each sprint (e.g., continuous integration helps to detect integration errors earlier and speed up releasing, refactoring improves product quality and thus minimizes adjustments for new features, collective code ownership reduces island knowledge and bottlenecks due to unnecessary specialization).

5. Service to the Product Owner

Coaching the Product Owner

- 5.1. ... identify at least three effective collaboration techniques that a Product Owner can use to work with the Scrum Team (e.g., engaging them in the

- shared purpose of their work, providing transparency of priorities, ensuring a shared understanding of product backlog items).
- 5.2. ... discuss at least three negative impacts that arise when the Product Owner applies excessive time pressure to the Development Team (e.g., quality is reduced, morale decreases, Definition of Done is not met).

6. Service to the Organization

Impediment Removal

- 6.1. ... discuss at least two ways that the Scrum Master assists the Scrum Team with responding to impediments (e.g., makes impediments visible, works with the Scrum Team to resolve impediments).
- 6.2. ... identify and explain at least three common organizational impediments outside the scope of a team that can affect the effectiveness of Scrum Teams (e.g., geographical distribution, people in multiple project teams, incentives and HR policies, no constructive safe-to-fail culture).

Coaching the Organization

- 6.3. ... describe one example of a major organizational design change implied by implementing Scrum (e.g., elimination of single-function groups, traditional career paths, or annual appraisals).
- 6.4. ... discuss why Scrum does not have a traditional project manager and what happens to traditional project management activities.
- 6.5. ... list at least three ways that traditional management changes in the Scrum workplace (e.g., management is not telling people what to do, people closest to the work make the majority of the decisions).
- 6.6. ... describe at least two stakeholder behaviors that support the Scrum Team's success and at least two behaviors that do not support the Scrum Team's success.
- 6.7. ... identify at least two organizational benefits of Scrum that are lost when the Scrum Team fails to adopt Scrum in its entirety.
- ... discuss at least two negative impacts to the organization when the Scrum Team has a lapse in dedication and discipline in using Scrum.

Program Teams

Updating our Certifications:

CSP and Progressive Course model

- Karim Harbott
- Carlton Nettleton
- Jason Tanner
- Andreas Schliep

Scrum Foundations and CSPO, CSPO Levels 2-3

- Peter Green
- Catherine Louis
- Jeff Patton
- Aaron Sanders

CSM

- Michael James
- Ron Jeffries
- Craig Larman
- Dave Prior
- Hubert Smits
- Vernon Stinebaker
- Mike Vizdos

Each group supported by Scrum Alliance staff Lisa Reeder and Erika Jones Massie.