

SCRUM ALLIANCE® CERTIFIED SCRUM PROFESSIONAL® - DEVELOPER (CSP®-D) Learning Objectives

August 2021



PURPOSE

This document describes the learning objectives (LOs) that must be covered in a Certified Scrum Professional® - Developer (CSP®-D) offering.

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 offering are based on:

- *Scrum Guide*, [scrumguides.org](https://www.scrumguides.org)*
- Manifesto for Agile Software Development, four values and 12 principles, [agilemanifesto.org](https://www.agilemanifesto.org)
- Scrum values, <https://www.scrumalliance.org/about-scrum/values>
- Scrum Alliance Scrum Foundations Learning Objectives
- Scrum Alliance Guide Level Feedback

Specific guiding resources are mentioned in the examples below.

SCOPE

Students attending a CSP-D offering should expect that each learning objective identified in this document will be covered.

The CSP-D learning objectives fall into the following categories:







1. Enabling a Culture of Technical Excellence
2. Catalyzing High-Performing Technology Organizations
3. Facilitating Environments for a Shared Understanding
4. Evolving Teams to Develop and Grow
5. Developing Self as an Agile Leader


Individual educators may choose to include ancillary topics. Ancillary topics presented in a CSP-D offering must be clearly indicated as such.


A note about Bloom's Taxonomy:

Bloom's-style learning objectives describe what the learner can do upon completing the offering. Please mentally start each learning objective with the following phrase: "Upon successful validation of the CSP-D learning objectives, the learner will be able to ... "

Bloom's style of Learning Objectives consist of six levels of learning:

-  **Knowledge**
-  **Comprehension**
-  **Application**
-  **Analysis**
-  **Synthesis**
-  **Evaluation**






The levels progress from lower-order to higher-order thinking skills. There are arguments whether "synthesis ,

 the capability to create something new, should actually count more than "evaluation . We distinguish between those high levels merely to indicate the different classes of capabilities and point to appropriate learning and validation strategies.











The level of each learning objective can be identified using the image designations above.

LEARNING OBJECTIVES




1 - Enabling a Culture of Technical Excellence

-  1.1 explain at least three tangible benefits of change measures toward higher operational excellence.
-  1.2 describe at least three aspects of technical coaching.
-  1.3 create a coaching agreement with one or multiple scrum teams.
-  1.4 practice at least three ways of coaching technical excellence with multiple teams on a technical topic.
-  1.5 propose at least three kinds of agile working agreements between teams and at least one action plan to uphold them.



2 - Catalyzing High-Performing Technology Organizations

-  2.1 integrate at least three design principles or patterns that enable emerging architectures.
-  2.2 apply at least one visual facilitation technique to model the behaviour of a system or product with users, clients, or stakeholders.
-  2.3 critique a legacy system using at least five criteria.
-  2.4 explain at least three agile design approaches to address typical challenges with legacy systems.
-  2.5 practice a continuous refactoring approach on a legacy system.
-  2.6 set up at least three aspects of an automated continuous integration pipeline.
-  2.7 explain at least three testing practices for agile development
-  2.8 demonstrate at least one testing practice for agile development beyond logical unit or component tests.
-  2.9 evaluate at least three, and practice at least one automated testing approach for system behavior.
-  2.10 outline at least three techniques to adopt continuous integration concepts beyond software.

3 - Facilitating environments for a shared understanding

-  3.1 explain at least three benefits of enabling a developer community of practice.
-  3.2 apply at least three supportive facilitation techniques to conduct a collaborative design session.
-  3.3 create at least one experiment to examine the impact of product work on customers, stakeholders, and/or the organization.

4 - Guiding Scrum Teams to Learn and Grow

-  4.1 outline at least three suitable learning formats for architecture and design principles.
-  4.2 demonstrate at least one learning format for architecture and design principles.

LEARNING OBJECTIVES

5 - Developing Self as an Agile Leader

- ✓ 5.1 compare and contrast leading others vs. demonstrating leadership.
- ↕ 5.2 apply at least one method to understand their own value system and relate it to the values and principles of the Manifesto for Agile Software Development.
- ⚙ 5.3 describe the importance of developer contribution to product exploration, customer discovery and experimentation.
- ↕ 5.4 apply a visual modelling technique for a value stream and select at least five improvement opportunities.

PROGRAM TEAM

CSP-D Program Team Members

- Andreas Schliep
- Axel Berle
- Bill Fairfield
- Björn Jensen
- Falk Kühnel
- Joe Justice
- Martin Salias
- Pascal Gugenberger
- Paul Ballew
- Paul Moore
- Rickard Jones
- Sherman Gomberg
- Rob Myers
- Veronica Ohl

*The *Scrum Guide* is © 2020 Ken Schwaber and Jeff Sutherland. Content from The *Scrum Guide* is used by permission of Ken Schwaber and Jeff Sutherland under the terms of the Creative Commons – Attribution – Share-Alike License v. 4.0. which is accessible at <https://creativecommons.org/licenses/by-sa/4.0/legalcode> and also described in summary form at <https://creativecommons.org/licenses/by-sa/4.0/>.