



PeopleCert SCRUM Master II

April 2021
Syllabus v2.0



E-mail: info@peoplecert.org
Website: www.peoplecert.org

Copyright © 2021 PeopleCert International Ltd.

All rights reserved. No part of this publication may be reproduced or transmitted in any form and by any means (electronic, photocopying, recording or otherwise) except as permitted in writing by PeopleCert International Ltd. Enquiries for permission to reproduce, transmit or use for any purpose this material should be directed to the publisher.

DISCLAIMER

This publication is designed to provide helpful information to the reader. Although every care has been taken by PeopleCert International Ltd in the preparation of this publication, no representation or warranty (express or implied) is given by PeopleCert International Ltd. as publisher with respect as to the completeness, accuracy, reliability, suitability or availability of the information contained within it and neither shall PeopleCert International Ltd be responsible or liable for any loss or damage whatsoever (indicatively but not limited to, special, indirect, consequential) arising or resulting of virtue of information, instructions or advice contained within this publication.)

1. Introduction

Scrum is a framework within which people can address complex adaptive problems, while productively and creatively delivering products of the highest possible value.

Scrum is a process framework for developing, delivering, and sustaining complex products that has been used to manage work on complex products since the early 1990s. Scrum is not a process, technique, or definitive method. 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 work techniques so that you can continuously improve the product, the team, and the working environment. The role of Scrum Master is crucial to ensuring a proper Scrum process is maintained throughout a project.

It's in this continuously evolving context, that PeopleCert designed a suite of Scrum Master qualifications that reflects the market need to close the skills-gap that many organizations are facing and support the realization of their business objectives and to improve communication, standardization, collaboration and automation for the delivery of quality software products better, faster and with a lower cost.

To reflect the need of this evolving work approach, the PeopleCert Scrum qualification has been structured as follows:

- **PeopleCert SCRUM Master I** (14-16 training hours) – Candidates get the essential knowledge needed by IT professionals in Agile methodologies and Scrum practices. It focuses on the fundamental principles and concepts of both Agile and Scrum.
- **PeopleCert SCRUM Master II** (14-16 training hours) – Candidates will enhance on all aspects of the Scrum framework through a variety of real-world scenarios and focused practical information related specifically to the Scrum Master.
- **PeopleCert SCRUM Product Owner stream** (coming soon) – Candidates will cover all aspects of the Scrum framework through a variety of real-world scenarios and focused practical information related specifically to the Scrum Product Owner.
- **PeopleCert SCRUM Developer** (coming soon) – Candidates will cover all aspects of the Scrum framework through a variety of real-world scenarios and focused practical information related specifically to the Scrum Developer.

The **PeopleCert SCRUM Master I** certification covers the fundamental **knowledge** required for a candidate to build their knowledge and skills regarding Scrum principles and practices. In addition, the **PeopleCert SCRUM Master II** certification (which is the next level of the qualification) covers more advanced skills, practices and knowledge about the Scrum framework.

The body of knowledge underlying these skills are presented in the official courseware provided by PeopleCert to accredited ATOs. The primary purpose of the syllabus is to provide a basis for accreditation of people involved with the Scrum framework. It documents the learning outcomes related to the qualification and describes the requirements a candidate is expected to meet to demonstrate that these learning outcomes have been achieved at the specific qualification level.

2. PeopleCert SCRUM Master II

2.1. Purpose of the SCRUM Master II Qualification

The purpose of this qualification level is to confirm that a candidate has sufficient knowledge, understanding and application of the Scrum framework and be able to work effectively with, or as a leading member of, a Scrum Team, analyzing and applying these skills and knowledge. The **PeopleCert SCRUM Master I** certification is highly recommended for the **PeopleCert SCRUM Master II** certification.

2.2. Target Group/Audience

This certification is the **second level** of the **PeopleCert SCRUM** qualification scheme and is aimed at anyone who wishes to become an efficient leading member in an environment that the Scrum framework is applied and requires candidates to have and demonstrate a solid **knowledge** and **understanding** of Scrum terms, principles, tools and practices, **as well as** demonstrate their **application** skills of how to use tools efficiently and effectively. The certification can also cater for candidates seeking personal certification.

This certification will provide the **advanced** level of knowledge to its holders and will certify that they have a solid understanding of the Scrum framework practices using various tools and are able to apply these in everyday work involving the required practices. The basic level of skills and knowledge is covered in the **PeopleCert SCRUM Master I** level of the qualification scheme provided by PeopleCert and a quick recap of these is provided in the **PeopleCert SCRUM Master II** material.

3. Learning Objectives

At this qualification level, candidates will be introduced to concepts, terms, principles and tools used in the Scrum framework and gain a deep understanding of Scrum best practices as well as a level of expertise that goes beyond a typical team leader or project manager through a unique combination of case studies, storytelling, and practical learning based on the real-life challenges that Scrum Masters face.

Holders of the **PeopleCert SCRUM Master II** certification, will be able to demonstrate their knowledge, understanding and practical application of:

- The true extent of what it means to think and act in an Agile way
- The mastery of the Scrum framework centered around the integration of roles, events, and Scrum artifacts, including and supporting the product backlog and sprint backlog
- The many soft skills necessary for the Scrum Master, such as servant leadership, facilitation, coaching and mentoring, problem solving, and removing impediments
- The Day-to-Day Work of the Scrum Master in a Sprint
- The most common frameworks for scaling Scrum projects with multiple teams and complex products

3.1. Qualification Scheme Level

Through the above learning objectives, candidates will demonstrate relevant knowledge skills in the following areas:

Main Topics

Introduction to Agile and Scrum
The Scrum Master Role
The Day-to-Day Work of the Scrum Master in a Sprint
An Introduction to Scaling Scrum

4. Examination

The **PeopleCert SCRUM Master II** certification exam is designed to validate a candidate's knowledge, and understanding, of the Scrum practices, as detailed above as well as how to be able to apply and analyze this knowledge in real situations faced by a Scrum Master.

The **SCRUM Master II** exam focuses on the following **four (4)** categories in the cognitive domain of **Bloom's taxonomy**¹ which is a reference for different levels of learning:

- **Knowledge**
- **Comprehension**
- **Application**
- **Analysis**

4.1. Assessment Approach

The assessment approach used for the **PeopleCert SCRUM Master II** certification focuses on the **four basic** categories of Knowledge, Comprehension, Application and Analysis.

Knowledge is defined as recalling previously learned material, from facts to theories and represents the lowest level of learning outcomes in the cognitive domain. Such learning outcomes are turned in assessment objectives that include **knowing and recalling** such as:

- Common and/or basic terms, definitions, concepts and principles
- Specific processes
- Processes, procedures and project management methods

Comprehension is the lowest level of understanding and entails the ability to grasp the meaning of the material taught, including some elements of interpretation, translation or estimation during the process. Such learning outcomes and in turn assessment objectives go beyond simply recalling information and may include:

- Understanding facts, concepts and principles
- Interpreting material (i.e. code, charts, graphs, text, diagrams)
- Justifying a process, procedure and method used

Application is a level where candidates need to combine their knowledge and understanding/comprehension on a subject and be able to create an abstraction. More specifically, candidates are expected to apply their knowledge and understanding so that abstractions, general principles, or methods to specific concrete

¹ The Bloom's taxonomy defines six (6) levels of learning in the **cognitive** domain (know, comprehend, apply, analyze, evaluate, create), which are both sequential and cumulative and move from the simple to the complex. In order to achieve the 6th level of learning, it must be ensured that the previous five levels have been mastered.

situations are made. Such learning outcomes and in turn assessment objectives go beyond simply recalling information and may include:

- Use ideas, principles and theories in new, specific and concrete situations
- Being able to choose appropriate methods and tools, apply principles, use a specific approach or identify the selection of options at a given situation
- Apply what is learnt into a new situation
- Apply rules, methods, concepts, principles, and theories
- Learning outcomes in this area require a higher level of understanding than those under comprehension

Analysis is the level that goes beyond application as the candidates need to be able to break down information into its component parts so that its organizational structure may be understood and to make inferences. More specifically, candidates need to break down, discriminate, diagram, detect, differentiate and illustrate which are all important tasks at this level of learning and include the previous levels of knowing, comprehending and applying. Such learning outcomes and in turn assessment objectives go beyond knowing, understanding and applying and may include:

- see patterns that they can use to analyze a problem
- develop divergent conclusions by identifying motives or causes
- make inferences
- find evidence to support generalizations
- identify parts, analyze the relationship between parts, and recognize the organizational principles involved

The assessment incorporates the above learning outcomes as it uses assessment objectives that cater for the above cognitive domain categories.

4.2. Entry Criteria/Training Requirements

For this examination there are no training requirements, though it is recommended that a candidate should hold the **PeopleCert SCRUM Master I** certification.

In order to be eligible for the **PeopleCert SCRUM Master II** level examination a candidate must be able to demonstrate knowledge and understanding of all Scrum terms, principles, processes, practices and be able to apply and analyze this knowledge in real situations assisting the Scrum framework application within a modern enterprise and it is recommended that the candidate has received related **Accredited Training** by a PeopleCert accredited training partner.

4.3. Examination Format

The following table details the examination format for the **PeopleCert SCRUM Master II** exam:

Delivery	Computer (web proctored or classroom)
Type	40 Multiple Choice Questions (MCQ) <i>Each question is awarded one (1) mark</i>
Duration	1 hour (60 minutes) <i>For non-native speakers or candidates with a disability, an additional 15 minutes of extra time is allowed.</i>
Pass Mark	70% (28 marks out of 40)
Invigilator / Supervisor / Proctor	Yes <i>Physical or Online Proctoring</i>
Open Book	No <i>No materials are allowed in the examination room</i>
Prerequisites	None <i>(PeopleCert SCRUM Master I certificate is recommended)</i>
Distinction	N/A
Certification validity	Perpetual

The tests are derived from a regularly updated question test bank (QTB) based on the test specification detailed below. Questions are used interchangeably among test sets. The overall difficulty level of each test is the same with any other test. A candidate is never assigned the same test in the case of multiple examination attempts.

5. Detailed Syllabus

The syllabus is structured into sections relating to the **major subject headings** and numbered with a single digit section number. The **recommended training hours, per Syllabus Category** are also provided in this table.

A total of **fourteen to sixteen (14-16) hours** of accredited training is **recommended**.

Category	Topic	Ref	Knowledge/Task Item
1 Introduction to Agile and Scrum	1.1 What Is Agile?	1.1.1	Define the term "Agile" as an approach for managing product development (including product value).
		1.1.2	Identify the four values of the Agile Manifesto.
		1.1.3	Identify the twelve principles of the Agile Manifesto.
		1.1.4	Discuss the need to 'do' and 'be' Agile in order to fully reap its benefits.
		1.1.5	Describe evidence of a successful Agile transformation.
	1.2 What is Scrum?	1.2.1	Describe Scrum as one of many Agile approaches.
		1.2.2	Explain the term 'Scrum' and why it is used.
		1.2.3	Explain the elements of the Scrum framework.
		1.2.4	Describe empiricism and lean thinking of Scrum theory (including divergent and convergent thinking).
		1.2.5	Recall the three pillars of scrum.
1.2.6		Identify the five core Scrum values.	
1.2.7		Recall the members of a Scrum team and know the difference between a team and a working group.	
1.2.8		Explain the concepts of cross-functionality and self-managing and what challenges these structures may raise for middle management.	
1.2.9		Describe the accountabilities of the Scrum team members.	
2 The Scrum Master Role	2.1 Core Competencies for a Scrum Master	2.1.1	Discuss the fundamental competencies required of a Scrum Master.
		2.1.2	Explain the need for a Scrum Master to be the Agile and Scrum Expert.
		2.1.3	Explain why a Scrum Master needs to be knowledgeable in other practices.
		2.1.4	Describe how a Scrum Master teaches others about Scrum.
		2.1.5	Explain the three coaching responsibilities of Scrum Master coaches.
		2.1.6	Describe the coaching skills required of a Scrum Master.
		2.1.7	Discuss the steps in the PLEASE coaching model.
		2.1.8	Explain the methods a Scrum Master uses to be a successful facilitator.

Category	Topic	Ref	Knowledge/Task Item
		2.1.9	Describe the skills of the Scrum Master when facilitating Scrum events.
		2.1.10	Describe the three levels of listening and the benefits of reaching the second and third levels.
		2.1.11	Explain the key aspects of effective communication for senders and recipients.
		2.1.12	Explain how to increase the effectiveness of team communication.
		2.1.13	Discuss problem solving skills and strategies.
		2.1.14	Describe the ways a Scrum Master can assist with conflict resolution.
		2.1.15	Explain the Scrum Master's role in helping individuals transition to a new way of working including moving through Bridges Transition Model.
	2.2 The Scrum Master as a Leader	2.2.1	Describe leadership qualities
		2.2.2	Explain the meaning of servant leadership.
		2.2.3	Describe the relationship of the core values to the Scrum Master's servant leadership role.
	2.3 Serving the Scrum Team	2.3.1	Describe the Scrum Master's service to the Scrum team.
		2.3.2	Explain what a Scrum Master needs to do in each of the stages of Tuckman's model for team development.
		2.3.3	Describe how a Scrum Master helps the Scrum team adapt to self-management.
		2.3.4	Explain the requirements for self-managing teams.
		2.3.5	Explain the signs of self-management.
		2.3.6	Describe the common misperceptions of self-managing teams
		2.3.7	Explain the topics and guidance for coaching the Scrum team.
		2.3.8	Describe types of interference and how the Scrum Master shields the team.
		2.3.9	Explain the meaning of impediments.
		2.3.10	Describe tactics for removing impediments.
	2.4 Serving the Product Owner	2.4.1	Identify the ways a Scrum Master serves the Product Owner.
		2.4.2	Recall the definitions for product and product goal.
		2.4.3	Recall the description and purpose of the product backlog.
		2.4.4	List the attributes of product backlog items.
	2.5 Serving the Organization	2.5.1	Describe the Scrum Master's service to the organization.
		2.5.2	Explain the Scrum Master's role as the Scrum framework authority.

Category	Topic	Ref	Knowledge/Task Item
		2.5.3	Describe how the Scrum Master acts as a change agent in an Agile/Scrum transition.
	2.6 What the Scrum Master Doesn't Do	2.6.1	Recall the behaviours Scrum Masters need to avoid.
3 The Day-to-Day Work of the Scrum Master in a Sprint	3.1 Preparing for the Sprint	3.1.1	Describe product backlog refinement.
		3.1.2	Explain the sizing techniques of 'ideal hours' and 'story points'.
		3.1.3	Describe the Planning Poker® approach to sizing product backlog items.
		3.1.4	Explain the use of sizes and triangulation for sizing product backlog items.
	3.2 The Sprint: Basic Rules	3.2.1	Explain the basic rules of Scrum as described in <i>The Scrum Guide</i> .
		3.2.2	Describe the 'definition of done' and its use.
		3.2.3	Explain the different time-boxed events in a sprint including their maximum timeframes.
	3.3 The Sprint: Sprint Planning	3.3.1	Recall the participants and three topics addressed during sprint planning.
		3.3.2	Describe the first topic of a sprint planning event including the participants
		3.3.3	Describe the second topic of a sprint planning event including participants, activities, inputs, and outputs.
		3.3.4	Explain the sprint backlog, its contents, and the sprint goal.
		3.3.5	Describe the third topic of a sprint planning event including participants, activities, inputs, and outputs.
		3.3.6	Explain the Scrum board, its contents, and use.
	3.4 The Sprint: Sprint Execution	3.4.1	Describe the participants in and the purpose of sprint execution.
		3.4.2	Describe the Developers' co-located workspace.
		3.4.3	Describe the daily Scrum including its purpose, participants, rules, and activities.
		3.4.4	Explain four possible questions asked during the daily Scrum.
		3.4.5	Describe the Developers' use of the Scrum board.
		3.4.6	Explain the sprint burn-down chart.
		3.4.7	Interpret the data displayed on a bar-style burn-down chart.
		3.4.8	Explain the concepts of the increment and incomplete work.
	3.5 The Sprint: Sprint Review	3.5.1	Describe the participants in and the purpose of the sprint review.

Category	Topic	Ref	Knowledge/Task Item
		3.5.2	Describe how the sprint review is carried out, including the roles of the Product Owner, Developers, and key stakeholders.
		3.5.3	Explain the tips for conducting, and the benefits of, the sprint review.
		3.5.4	Describe Release Planning
	3.6 The Sprint: Sprint Retrospective	3.6.1	Describe the sprint retrospective including its participants, purpose and timing
		3.6.2	Explain what happens during the sprint retrospective.
		3.6.3	Explain the concept of improvement stories.
		3.6.4	Explain the concept of the improvement board.
4 An Introduction to Scaling Scrum	4.1 Scrum@Scale®	4.1.1	Recall the definition of the Scrum@Scale framework and understand why an organization may not choose to scale.
		4.1.2	Recall the two basic cycles of the Scrum@Scale framework and the points at which they intersect.
		4.1.3	Recall the composition of the Scrum@Scale teams.
		4.1.4	Recall the scaled events that are part of Scrum@Scale.
	4.2 Scaling Scrum with Nexus™	4.2.1	Recall the definition of the Nexus framework.
		4.2.2	Recall the composition of the Nexus teams.
		4.2.3	Recall the scaled events that are part of the Nexus framework.
	4.3 Scaled Agile Framework (SAFe)®	4.3.1	Recall the definition of the Scaled Agile Framework (SAFe)
		4.3.2	Recall the composition of the Scaled Agile Framework (SAFe) teams
		4.3.3	Recall the seven core competencies Scaled Agile Framework (SAFe)

Scrum@Scale is a registered trademark of Scrum Inc.

SAFe and Scaled Agile Framework are registered trademarks of Scaled Agile, Inc.

6. Test Specification

The **PeopleCert SCRUM Master II** examination will consist of **four (4)** sections with the following structure:

Category	Description	Exam (%)
1.0	Introduction to Agile and Scrum	17.5%
2.0	The Scrum Master Role	42.5%
3.0	The Day-to-Day Work of the Scrum Master in a Sprint	37.5%
4.0	An Introduction to Scaling Scrum	2.5%
	Total	100.0%

7. Bibliography

1. Beck, K., et al. (2001) The Agile Manifesto. Agile Alliance. <http://agilemanifesto.org/>
2. Belknap, Chris. "How does a Scrum Master help a team self-organize?". LinkedIn.com. Accessed August 17, 2020. <https://www.linkedin.com/pulse/how-does-scrum-master-help-team-self-organize-chris-belknap/>
3. Bridges, William. "Bridges Transition Model." William Bridges Associates - Transition Management Leaders, 1988.
4. Cohn, Mike. *Agile Estimating and Planning*. Upper Saddle River, NJ: Prentice Hall, 2005-2011.
5. "Mountain Goat Software." "Why the Fibonacci Sequence Works Well for Estimating." Accessed August 23, 2020. <https://www.mountaingoatsoftware.com/blog/why-the-fibonacci-sequence-works-well-for-estimating>
6. "How to Prevent Estimate Inflation". Mountain Goat Software. Accessed August 23, 2020. <https://www.mountaingoatsoftware.com/blog/how-to-prevent-estimate-inflation>
7. Corry, Paddy. "5 Essential Facilitation Skills for Scrum Masters and How to Develop Them". Serious Scrum. Last modified September 30, 2019. Accessed February 29, 2020. <https://medium.com/serious-scrum/5-essential-facilitation-skills-for-scrum-masters-and-how-to-develop-them-19088080354c>
8. de Jonge, Marty. "Self-Organization versus Self-Management". Medium.com. Accessed December 10, 2020. <https://medium.com/serious-scrum/self-organization-versus-self-management-f7fddf62851>
9. Graffius, Scott M. "Use Tuckman's Model of Team Dynamics". AgileScrumGuide.com. Accessed August 19, 2020. <https://agilescrumguide.com/blog/files/Use-Tuckmans-Model-of-Team-Dynamics.html>
10. In-Tuition. "Trainer Skills and the Impact on Productivity". In-Tuition – Practical Management Skills. Accessed January 24, 2020. <http://www.practical-management-skills.com/trainer-skills.html>
11. Indeed. "Problem Solving Skills: Definitions and Examples". Indeed.com. Accessed August 20, 2020. <https://www.indeed.com/career-advice/resumes-cover-letters/problem-solving-skills>
12. Innovel. "Improve Measurably with Improvement Stories". Innovel.net. Accessed August 31, 2020. <https://innovel.net/improve-measurably-with-improvement-stories/>
13. Kilmann, Dr. Ralph H. The Thomas-Kilmann Conflict Mode Instrument. Accessed August 13, 2020. <https://kilmanniagnostics.com/overview-thomas-kilmann-conflict-mode-instrument-tki/>
14. Knowledge Hut. "Scrum Master". Knowledgehut.com. Accessed August 18, 2020. <https://www.knowledgehut.com/tutorials/scrum-tutorial/scrum-master>
15. "12 Common Mistakes of the Scrum Master and the Remedies". Accessed August 18, 2020. <https://www.knowledgehut.com/blog/agile/12-common-mistakes-scrum-master-and-remedies>
16. Kotter, John P., Dan S. Cohen. *The Heart of Change: Real-Life Stories of How People Change Their Organizations*. Boston, Massachusetts, USA: Harvard Business School Press, August 1, 2002.

17. Lacey, Mitch. "Adding the Fourth Question to the Daily Standup." Mitch Lacey & Associates, Inc. Modified December 5, 2010. Accessed January 12, 2021. <https://www.mitchlacey.com/blog/scrum-adding-the-fourth-question-to-the-daily-standup/>
18. Lacey, Mitch. *The Scrum Field Guide: Agile Advice for Your First Year and Beyond*. Boston, MA: Pearson Education, 2017.
19. Lynn, Rachaelle. "What Is a Self-Organizing Team?" Planview. Accessed December 4, 2019. <https://www.planview.com/resources/articles/what-is-self-organizing-team/>
20. Morris, David. *Scrum in Easy Steps – an ideal framework for agile projects*. UK: In Easy Steps Limited, 2017.
21. National Soft Skills Association™. "The Soft Skills Disconnect." NationalSoftSkills.org. Modified February 13, 2015. Accessed January 11, 2020. <https://www.nationalsoftskills.org/the-soft-skills-disconnect/>
22. Overeem, Barry. "Characteristics of a Great Scrum Team". InfoQ. Last modified April 15, 2016. Accessed February 27, 2020. <https://www.infoq.com/articles/great-scrum-team/>
23. Overeem, Barry. Scrum.org. "The Scrum Master as a Servant-Leader". Last modified July 20, 2015. Accessed August 12, 2019. <https://www.scrum.org/resources/blog/scrum-master-servant-leader>
24. Overeem, Barry. "The Scrum Master as an Impediment Remover." Accessed September 23, 2020. <https://www.scrum.org/resources/blog/scrum-master-impediment-remover>
25. Partogi, Joshua. "Scrum Master: The Master of the Art of Facilitation". Scrum.org. Last modified August 19, 2018. Accessed February 29, 2020. <https://www.scrum.org/resources/blog/scrum-master-master-art-facilitation>
26. Ravlani, Kamlesh. "The Scrum Master as a Coach". Agile for Growth. Accessed September 2, 2020. <https://agileforgrowth.com/blog/scrummaster-coach/#Skills-Coach>
27. Ravlani, Kamlesh. "7 Ways the Scrum Master Can Improve Scrum Team Communication". Agile for Growth. Last modified April 17, 2019. Accessed February 29, 2020. <https://agileforgrowth.com/blog/scrummaster-coach/>
28. Raza, A. and Hammad Majeed. "Issues and Challenges in Scrum Implementation." International Journal of Scientific & Engineering Research, Issue 3, Number 8 (2012): 2229-5518.
29. Rubin, Kenneth. *Essential Scrum: A Practical Guide to the Most Popular Agile Process*. Boston, MA: Pearson Education, 2013.
30. Scaled Agile, Inc. "Download the Big Picture" Scaledagile.com. Accessed December 18, 2020. <https://www.scaledagileframework.com/posters/>
31. Scaled Agile, Inc. "What is SAFe?" Scaledagile.com. Accessed December 18, 2020. <https://www.scaledagile.com/enterprise-solutions/what-is-safe/>
32. Smith, Richard, David King, Ranjit Sidhu, Dan Skelsey, and APMG. *The Effective Change Manager's Handbook: Essential Guidance to the Change Management Body of Knowledge*. London and New York: Kogan Page, 2014.

33. Schwaber, Ken and Jeff Sutherland. *The Scrum Guide™, the Definitive Guide to Scrum: The Rules of the Game*. 2017. Accessed August 12, 2019.
34. Schwaber, Ken and Jeff Sutherland. *The Scrum Guide™, the Definitive Guide to Scrum: The Rules of the Game*. 2020. Accessed December 9, 2020.
35. Schwaber, Ken and Scrum.org. *The Nexus™ Guide: The Definitive Guide to Scaling Scrum with Nexus*. Scrum.org, 2021. Licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.
36. Scrum Forum. "Stakeholders constantly change the requirements". Scrum.org. Accessed August 18, 2020. <https://www.scrum.org/forum/scrum-forum/29944/stakeholders-constantly-change-requirements>
37. "Scrum Master is NOT responsible for". Accessed August 18, 2020. <https://www.scrum.org/forum/scrum-forum/25667/scrum-master-not-responsible>
38. Scrum Guide Refresh July 2016 - Scrum Pulse Episode #14. Accessed August 19, 2020. <https://www.youtube.com/watch?v=0hRZffDD1ec>
39. Scrum Guides. "Changes between 2010 and 2011 Scrum Guides". Accessed August 20, 2020. <https://www.scrumguides.org/revisions.html>
40. Scrum.org. *Nexus™ Guide, The Definitive Guide to scaling Scrum with Nexus: The Rules of the Game*. 2018. <https://www.scrum.org/resources/scaling-scrum> Accessed 8/31/20
41. The Evidence-Based Management Guide: Measuring Value to Enable Improvement and Agility. September 2020. <https://www.scrum.org/resources/evidence-based-management-guide>
42. Licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.
43. Shift314. "Doing Agile vs. Being Agile". Agilitrix. Last modified April 26, 2016. Accessed February 29, 2020. <https://shift314.com/2016/04/doing-agile-vs-being-agile/>
44. Software Testing Help. "Self-Sufficient Scrum Teams: How to Create a Self-Sufficient Team?". Softwaretestinghelp.com. Accessed August 17, 2020. <https://www.softwaretestinghelp.com/self-sufficient-scrum-teams/>
45. Surdek, Steffan. "Three Common Misunderstandings of Self-Organized Teams." *Forbes*. Last modified December 20, 2016. Accessed February 27, 2020. <https://www.forbes.com/sites/forbescoachescouncil/2016/12/20/three-common-misunderstandings-of-self-organized-teams/#62aedd4b195e>
46. Sutherland, Jeff and Scrum Inc. *The Scrum@Scale® Guide, The Definitive Guide to Scrum@Scale: Scaling That Works*. Scrum.org, 2021. Accessed December 18, 2020. Licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.
47. Thomas, Dr. Kenneth W. and Dr. Ralph H. Kilmann. "An Overview of the TKI Assessment Tool," Kilmann Diagnostics. Accessed August 13, 2020. <https://kilmandiagnostics.com/overview-thomas-kilmann-conflict-mode-instrument-tki/>

48. Tomić, Smilja. "Active Listening as a Tool for Scrum Masters." LinkedIn. Last modified December 18, 2017. Accessed February 27, 2020. <https://www.linkedin.com/pulse/active-listening-tool-scrum-masters-smilja-tomi%C4%87>
49. Watts, Geoff. *Scrum Mastery - From Good to Great Servant-Leadership*. Cheltenham, UK: Inspect and Adapt Ltd., 2013.

8. Glossary

Term	Definition
acceptance criteria	Acceptance criteria are a formal list of external quality characteristics specified by the Product Owner that give guidance about whether a product backlog item has been successfully developed and will, therefore, be accepted by the customer.
adaptation	One of the three pillars of empirical process control and, with the other two – transparency and inspection – provide the feedback necessary to make continual improvements to the product being developed.
affinity estimation	A quick and easy technique used by the Development Team to quickly estimate a large number of user stories using categories that indicate relative size; example: t-shirt sizes (XS, S, M, L, XL)
Agile	A project management style focused on the early delivery of business value, continuous improvement, scope flexibility, team input, and delivering well-tested products that reflect customer needs.
Agile Manifesto	The Agile Manifesto is a document written in 2001 by seventeen independent-minded software practitioners in Snowbird, Utah that is comprised of four foundational values and 12 supporting principles that guide the development and delivery of high-quality, working software.
artifact	Scrum artifacts provide key information the Scrum team and the stakeholders need to be aware of to understand the product under development, the activities being planned, and the activities conducted during the project. Artifacts defined by Scrum are specifically designed to maximize the transparency of key information, so everyone has the same understanding of the artifact. The following artifacts are defined in Scrum: the product backlog, the sprint backlog, and the increment.
bug list (or defect board)	A separate task board that includes all errors, bugs, and problems
capacity	<ol style="list-style-type: none"> 1. The quantity of resources available to perform useful work 2. A concept used to help establish a WIP limit by ensuring that we only start work to match the available capacity to complete work
co-location	Co-location is having all Scrum core team members located in the same workplace to leverage the advantages of better coordination, problem-solving, knowledge sharing, and learning.
complex systems (in the context of organizational change)	Complex systems are comprised of a large number of elements that interact in nonlinear ways, and their behaviour is determined by the nature of these interactions. They are open systems in which a stable state is not desirable. These systems are often referred to as complex adaptive systems because unpredictable, novel characteristics may emerge that may or may not be desirable. (Blignaut, 2019)
component team	This is a team responsible for an architecture building block such as one of the layers of a feature. One component team may be responsible for the user interface layer, while other teams are responsible for the business logic layer and the data layer.
continuous quality	This is a systematic approach for identifying and addressing problems as early as possible in a development life cycle. Early and continued validation is essential and can be achieved through a variety of approaches such as peer reviews, test driven development, and unit testing, among others. Often referenced in conjunction with continuous integration/continuous delivery. (Dowd, 2018)
cross-functional	Having teams with all the skills and experience necessary to meet their common purpose and goals
daily Scrum	The daily Scrum is a 15-minute time-boxed event for the Development Team that is held at the same time and place every day of the sprint. During this time, the Development Team plans work for the next 24 hours. The daily Scrum is a synchronization, inspection, and adaptive planning activity that optimizes team collaboration and performance. May also be known as the daily stand-up.
decomposition	Decomposition is a tool whereby high-level tasks are broken down into lower level, more detailed tasks. The user stories are decomposed into tasks by members of the Scrum team. Prioritized product backlog user stories should be sufficiently decomposed to a level that provides the Scrum team with adequate information to create deliverables from the tasks mentioned in the task list.

Term	Definition
DEEP	An acronym coined by Roman Pichler and Mike Cohn for remembering a set of criteria used to evaluate the quality of a product backlog. The criteria are d etailed appropriately, e mergent, e stimated, and p rioritized.
definition of done	This is an agreed-upon list of the activities deemed necessary to get a product increment, usually represented by a user story, to a done state by the end of a sprint. The definition of done ensures everyone on the team knows exactly what is expected of everything the team delivers. It ensures transparency and quality are fit for the purpose of the product and organization.
definition of ready	The definition of ready involves creating clear criteria that a user story must meet before being accepted into an upcoming iteration. This is typically based on the INVEST matrix.
Developers	Developers are the people in the Scrum team who are committed to creating any aspect of a usable increment each sprint. (Previously referred to as the Development Team.)
development work	The work required to create a releasable increment of product every sprint
empiricism	Empiricism asserts that knowledge comes from experience and making decisions based on what is observed. Empiricism has three pillars: transparency, inspection, and adaptation.
empirical process control	This is a process control type in which only the past is accepted as certain and in which decisions are based on observation, experience and experimentation.
epic	A large and unrefined user story in the product backlog that is written in the initial stages of the project and can span an entire release
estimation	See sizing.
feature	A collection of related user stories that is used for products that are large and/or complex
feature team	This is a team of Developers that implements end-user functionality end-to-end; contrasts with a component team
Fibonacci sequence	The Fibonacci scale consists of a sequence of numbers used for estimating the relative size of user stories in points, resulting in reduced complexity, effort, and doubt when determining the time required for product backlog items. Each number is the sum of the two preceding ones (1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144...). It was named after the Italian mathematician Leonardo of Pisa.
fixed-date release	A fixed-date release is one that must be delivered on a known future date; the scope of the release, and possibly the cost, needs to be flexible; and it contrasts with the fixed-scope release.
fixed-scope release	A fixed-scope release is one that must have a specific set of features, include the date on which the features are delivered and/or flexible costs, and contrasts with fixed-date release.
functional criteria	Defines the functions the PBI will introduce, and identifies specific user tasks, functions, or business processes that must be in place
ideal time	The time estimated to complete a user story in a perfect situation and in the following conditions: <ul style="list-style-type: none"> • Without any interruption, risk events, or process wastes such as waiting • With all the knowledge, tools, skills, team support, and requirements information available
increment	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.
incremental development	Nearly all Agile teams favour an incremental development strategy. In an Agile context, this means each successive version of the product is usable, and each builds upon the previous version by adding user-visible functionality. These are called "vertical" increments (that is, the difference between successive product versions), as opposed to the opposite strategy that successively delivers complete technical components; for instance, creating a database schema, then building business rules on top of that, and only then implementing a UI.
information radiator	Information radiator is a popular term invented by Alistair Cockburn that is used to describe any artifact that conveys project information and is publicly displayed in the workspace or surroundings. Information radiators are very popular in the Agile world, and they are an essential component of visual management.
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 it gets in the way of the work. Inspections are most beneficial when diligently performed by skilled inspectors at the point of work.

Term	Definition
INVEST	<p>The acronym INVEST stands for a set of criteria used to assess the quality of a user story. If the story fails to meet one of these criteria, the team may want to reword it.</p> <p>The criteria are:</p> <ul style="list-style-type: none"> • Independent • Negotiable • Valuable • Estimable • Small • Testable
iterative development	Agile projects are iterative insofar as they intentionally allow for "repeating" software development activities, and for potentially "revisiting" the same work products (the phrase "planned rework" is sometimes used; refactoring is a good example). They are iterative in a third, less-essential sense, in being most often structured around a series of iterations of fixed calendar length.
legacy bugs	The errors identified in testing, but never corrected for the expediency of releasing work on time
kanban	The <i>kanban</i> method is a product and systems thinking approach to improving the delivery of services to customers, as well as the environment in which those delivering the service operate and interact with one another.
minimally viable product (MVP)	A minimum viable product is, as Eric Ries said, the "version of a new product that allows a team to collect the maximum amount of validated learning about customers with the least effort."
MoSCoW method	"The MoSCoW method is a prioritization technique used in management, business analysis, project management, and software development to reach a common understanding with stakeholders on the importance they place on the delivery of each requirement; it is also known as the MoSCoW prioritization or MoSCoW analysis. The term MoSCoW itself is an acronym derived from the first letter of each of four prioritization categories (must have, should have, could have, and won't have), with the interstitial Os added to make the word pronounceable. While the Os are usually in lower-case to indicate that they do not stand for anything, the all-capitals MOSCOW is also used."
naïve technical debt	This is a form of technical debt that accrues due to irresponsible behaviour or immature practices on the part of the people involved; contrasts with strategic technical debt, or unavoidable technical debt.
<i>niko-niko</i> calendar	This is an information radiator where each team member expresses their mood by drawing or attaching an emoticon on the calendar at the end of the day. The Scrum Master uses the information to find the root causes of some unseen problems. The whole Scrum team uses the <i>niko-niko</i> chart in the sprint retrospective to plan for improvements.
non-functional criteria	These define the conditions that the PBI must meet, and identify specific non-functional conditions that the implementation must meet, such as design elements
osmotic communication	The accidental overhearing of background information that may later end up being important
release parking-lot chart	A parking-lot chart is a powerful method for compressing a great deal of information into a small space. Each theme or grouping of user stories contains a large box that is displayed for the team to see the release progress.
performance criteria	These are explicitly included where speed of response, availability, or scalability is critical to the acceptance of an individual PBI.
planning onion, Cohn's	Mike Cohn refers to multi-layer planning as the planning onion. At different layers of this planning onion, the project team focuses on varying level of details at different stages of the project. The planning onion helps teams choose the right level of planning for each timeframe for which they are planning, and has six levels – strategy, portfolio, release, iteration, daily, and continuous.
Planning Poker®	This is an approach used by Agile teams to establish the size of product backlog items that considers the relative effort to complete the work. It balances group thinking and individual thinking.
portfolio planning (portfolio management)	An activity for determining which products (or projects) to work on, in which order, and for how long

Term	Definition
potentially shippable product	These are results that are completed to a high degree of confidence and represent work of good quality that is potentially shippable to end customers at the end of a sprint. Being potentially shippable does not mean the results will actually be delivered to customers. Shipping is a business decision; potentially shippable is a state of confidence.
predictive development	This is a style of development that is phase-based and sequential. It assumes that all the necessary correct information has been made available from the very start of a project
product backlog	This is a prioritized list of new features, changes to existing features, bug fixes, infrastructure changes, or other activities that a team may deliver to achieve a specific outcome of a product. The Product Owner is the sole person responsible for managing the product backlog.
product backlog refinement	Backlog grooming is when the Product Owner and some, or all, of the rest of the team refine the backlog on a regular basis to ensure the backlog contains the appropriate items, that they are prioritized, and that the items at the top of the backlog are ready for delivery.
product backlog item	<ol style="list-style-type: none"> 1. An item such as a feature, defect, or occasionally technical work that is valuable from the Product Owner's perspective 2. An item in the product backlog
Product Owner	The Product Owner is a single individual who is accountable for maximizing the value of the product resulting from the work of the Scrum team and for effective product backlog management.
product roadmap	A description of the incremental nature of how a product will be built and delivered over time, along with the important factors that drive each individual release. Useful when developing a product that will have more than one release
product vision	This is a brief statement of the desired future state that would be achieved by developing and deploying a product. A good vision should be simple to state and provide a coherent direction to the people who are asked to realize it.
project charter	This is a document issued by the project initiator or sponsor that formally authorizes the existence of a project, and provides the project manager with the authority to apply organizational resources to project activities. (Brown, A.S., 2005)
release	<ol style="list-style-type: none"> 1. A combination of features that when packaged together make for a coherent deliverable to customers or users 2. A version of a product that is promoted for use or deployment. Releases represent the rhythm of business-value delivery and should align with defined business cycles.
release backlog	A collection of product backlog items (PBIs) that have been selected or will be selected for the current release work effort
release burndown chart	A graphical, real-time picture of the work that is remaining in the release and allows for the easy visualization of whether the release is on track to reach completion on time and in scope
release plan	<ol style="list-style-type: none"> 1. This is the output of release planning. On a fixed-date release, the release plan will specify the range of features available on the fixed future date. On a fixed scope release, the release plan will specify the range of sprints and costs required to deliver the fixed scope. 2. This is a plan that communicates, to the level of accuracy that is reasonably possible, when the release will be available, what features will be in the release, and how much will it cost.
release planning	The release planning schedule is one of the key outputs of the conduct release planning process. A release planning schedule states which deliverables are to be released to the customers, along with planned intervals, and dates for releases. There may not be a release scheduled at the end of every sprint iteration.
return on investment (ROI)	Return on investment (ROI), when used for project justification, assesses the expected net income to be gained from a project. It is calculated by deducting the expected costs or investment in a project from its expected revenue and then dividing this (net profit) by the expected costs to get a return rate.
risk	Risk is defined as an uncertain event or set of events that can affect the objectives of a project and may contribute to its success or failure.
root cause analysis	Root cause analysis (RCA) is a systematic process for identifying the "root causes" of problems or events and is also an approach for responding to them. RCA is based on the basic idea that effective management requires more than merely "putting out fires" for problems that develop, but also finding a way to prevent them.
Scrum	This is a lightweight framework, in use since the early 1990s, that helps people, teams, and organizations generate value through adaptive solutions for complex problems.

Term	Definition
Scrum artifacts	Provide key information that Scrum teams and stakeholders need to be aware of for understanding the products under development, the activities being planned, and the activities performed during a given project
Scrum board	A Scrum board is a tool used by the Scrum team to plan and track progress during each sprint. The Scrum board contains four columns to indicate the progress of the estimated tasks for the sprint: a “to do” column for tasks not yet started, an “in progress” column for the tasks started but not yet completed, a “testing” column for tasks completed but in the process of being tested, and a “done” column for the tasks that have been completed and successfully tested.
Scrum event	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. These include the sprint, sprint planning, daily Scrum, sprint review, and sprint retrospective. Each event in Scrum is a formal opportunity to inspect and adapt something. These events are specifically designed to enable critical transparency and inspection.
Scrum Master	The Scrum Master works with the Product Owner, Development Team, and other involved parties to ensure artifacts maintain transparency. They ensure the Scrum events take place and that attendees understand their purpose. They also teach the Development Team to keep the events within the time-box.
Scrum reference card	A chart that works as a basic tool for finding the right approach and mix of development practices
Scrum of Scrums (SoS)	This is a technique to scale Scrum up to large groups (over a dozen people), consisting of dividing the groups into Agile teams of five to 10. Each daily Scrum within a sub-team ends by designating one member as the "ambassador" to participate in a daily meeting with the ambassadors from other teams, called the Scrum of Scrums.
Scrum team	This refers to the team that is comprised of the Product Owner, Scrum Master, and Developers, and which is accountable for creating a valuable, useful increment every sprint. The team is responsible for all product-related activities from stakeholder collaboration, verification, maintenance, operation, experimentation, research, and development.
Scrum values	When the values of commitment, courage, focus, openness, and respect are embodied and lived by the Scrum team, the Scrum pillars of transparency, inspection, and adaptation come to life and build trust for everyone. The Scrum team members learn and explore those values as they work with the Scrum events, roles, and artifacts. The successful use of Scrum depends on people becoming more proficient in living these five values.
self-managing	This is a property of a team that indicates the team internally decides who does what, when, and how without direction or control from an outside party.
servant leader	Servant leaders employ listening, empathy, commitment, and insight while sharing power and authority with team members. Servant leaders are stewards who achieve results by focusing on the needs of the team. This style is the embodiment of the Scrum Master role.
sizing	Assigning a value to represent the effort required to complete a product backlog item to the definition of done
SMART	SMART is a mnemonic/acronym, giving criteria for guidance when setting objectives; for example, in project management, employee-performance management, and personal development. The letters S and M generally mean specific and measurable. Possibly the most common version has the remaining letters referring to attainable (or achievable), relevant/ realistic, and timely.
sprint	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. Sprints have consistent durations throughout a development effort and 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.
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.
sprint burn-down and burn-up charts	Burn-down charts and burn-up charts track the amount of output (in terms of hours, story points, or backlog items) a team has completed across an iteration or a project.

Term	Definition
sprint demo	<ol style="list-style-type: none"> 1. An activity of a sprint review where the completed (done) product backlog items are demonstrated with the goal of promoting an information-rich discussion between the Scrum team and other sprint review participants 2. A term that is frequently used synonymously to refer to the entire sprint review
sprint execution	A stage in which development work gets done to meet a sprint goal
sprint goal	It is the single objective for the sprint and represents the commitment for the sprint backlog. The sprint goal is created during the sprint planning event and added to the sprint backlog. It creates coherence and focus, encouraging the Scrum team to work together rather than on separate initiatives.
sprint planning	The sprint planning meeting is conducted at the beginning of a sprint as part of the create-a-sprint-backlog process. It is time-boxed to eight hours for a one-month sprint and addresses three topics: why the sprint is valuable; what can be done during the sprint; and how the chosen work will get done.
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 at most a three-hour meeting for one-month sprints. For shorter sprints, the event is usually shorter. The Scrum Master ensures the event takes place and that attendees understand its purpose.
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.
stakeholders	Stakeholders are people who affect or are affected by your project. Internal stakeholders are within your company or organization; they could be from the legal, sales, marketing, management, procurement, and/or any other division of your company. External stakeholders could be investors or users.
strategic technical debt	This is a form of technical debt that is used as a tool to help organizations better quantify and leverage the economics of important, often time-sensitive, decisions. Sometimes taking on technical debt for strategic reasons is a sensible business choice. Contrasts with naive technical debt, which is unavoidable technical debt
Story mapping	Story mapping consists of ordering user stories along two independent dimensions. The "map" arranges user activities along the horizontal axis in a rough order of priority (or "the order in which you would describe activities to explain the behaviour of the system"). Down the vertical axis, it represents the increasing sophistication of the implementation.
story point	Story points are a unit of measure for expressing an estimate of the overall effort that will be required to fully implement a product backlog item or any other piece of work. Product backlog items such as stories, are usually sized/estimated with story points. This is usually based on the Fibonacci sequence or t-shirt sizes.
sustainable pace	The team aims for a work pace they would be able to sustain indefinitely. This entails a firm refusal of what is often considered a "necessary evil" in the software industry – long work hours, overtime, or even working nights or weekends. As such, this "practice" is really more of a contract negotiated between the team and their management.
swarming	An approach that involves teams working together on their highest priority work
task board	An information radiator used during the sprint execution to communicate the progress and flow of task-level work within a sprint and ensures efficient dissemination of relevant information to the entire team
technical debt	Technical debt (also referred to as design debt or code debt) refers to the work that teams prioritize lower, omit, or do not complete as they work towards creating the primary deliverables associated with the project's product. Technical debt accrues and must be paid in the future.
theme	A large business functionality that acts as a category or grouping of user stories (ranging from epics to small user stories)
thrashing	An approach that involves including too much work in progress
three pillars	Scrum uses the empirical process control that relies on the three pillars of transparency, inspection, and adaptation.

Term	Definition
timebox/timeboxing	A timebox is a previously agreed period of time during which a person or a team works steadily toward the completion of some goal. Rather than allow work to continue until the goal is reached and then evaluating the time taken, the timebox approach consists of stopping work when the time limit is reached and evaluating what was accomplished.
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 that observers share a common understanding of what is being seen. For example, those performing the work and those inspecting the resulting increment must share a common definition of "done".
triangulation	Sizing of product backlog items is performed by comparing a chosen user story to one that is smaller and then one that is larger. Triangulation helps to ensure appropriate sizing and consistency, and to prevent size inflation.
t-shaped skills	Skills that describe specific attributes of desirable workers
unavoidable technical debt	This is a form of technical debt that is usually unpredictable and unpreventable and accrues through no fault of the team building the product; contrasts with the naïve technical debt, or strategic technical debt.
user stories	In consultation with the customer or Product Owner, the team divides up the work to be done into functional increments called "user stories." Each user story is expected to yield, once implemented, a contribution to the value of the overall product, irrespective of the order of implementation; these and other assumptions as to the nature of user stories are captured by the INVEST formula.
velocity	At the end of each iteration, the team adds up effort estimates that were associated with the user stories completed during that iteration. This total is called velocity. Knowing velocity, the team can compute (or revise) an estimate of how long the project will take to complete, based on the estimates associated with the remaining user stories and assuming that the velocity over the remaining iterations will remain approximately the same. This is generally an accurate prediction, even though rarely a precise one.
visual management	This is the practice of using information visualization techniques to manage work. A simple example is using sticky notes on a wall to manage a list of tasks, a better (and more complex) example is kanban.
voice of customer	The voice of the customer (VoC) can be referred to as the explicit and implicit requirements of the customer that must be understood prior to the designing of a product or service. The Product Owner represents the voice of the customer.

GLOSSARY RESOURCES

Agile Alliance. "Agile Glossary." Last modified: 2011. Accessed: October 30, 2019.

<https://www.agilealliance.org/agile101/agile-glossary/>

Blignaut, Sonja. "7 Implications of seeing organizations as complex systems," Agile & Change. August 28, 2019.

<https://agileandchange.com/7-implications-of-seeing-organisations-as-complex-systems-996fd2398d58>

Brown, A. S. (2005). The charter: selling your project. Paper presented at PMI® Global Congress 2005—North America, Toronto, Ontario, Canada. Newtown Square, PA: Project Management Institute.

Dowd, Ryan. "Continuous Quality". TechTarget.com. Last updated January 2018. Accessed April 7, 2021.

<https://searchsoftwarequality.techtarget.com/definition/continuous-quality>

Rubin, Kenneth. *Essential Scrum: A Practical Guide to the Most Popular Agile Process*. Boston, MA: Pearson Education, 2013.

Schwaber, Ken and Jeff Sutherland. *The Scrum Guide™, The Definitive Guide to Scrum: The Rules of the Game*.

Scrum.org, November 2020. <https://www.scrum.org/resources/scrum-guide>

Volkerdon.com. "Agile Glossary." Accessed October 30, 2019. <https://www.volkerdon.com/pages/agile-glossary>

This page is left intentionally blank

