



Learning Outcomes

Agile Delivery

Core & Specialized Credentials

International Certification Alliance

Version 1.0 — August 2026



LICENSING INFORMATION

The work in this document was facilitated by the International Certification Alliance (ICA) and done through the contributions of various experts and practitioners. These learning objectives are intended to help the growing Tomorrow-Ready™ community worldwide.

This work is licensed under the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/> or send a letter to:

Creative Commons,
P.O. Box 1866,
Mountain View, CA 94042, USA

YOU ARE FREE TO:

Share — copy and redistribute the material in any medium or format

UNDER THE FOLLOWING TERMS:

Attribution: You must give appropriate credit to The International Certification Alliance (ICA), provide a link to the license, and indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.

NonCommercial: You may not use the material for commercial purposes.

NoDerivatives: If you remix, transform, or build upon the material, you may not distribute the modified material.

NOTICES:

You do not have to comply with the license for elements of the material in the public domain or where your use is permitted by an applicable exception or limitation.

No warranties are given. The license may not give you all of the permissions necessary for your intended use. For example, other rights such as publicity, privacy, or moral rights may limit how you use the material.



TABLE OF CONTENTS

LICENSING INFORMATION	2
TABLE OF CONTENTS	3
HOW TO READ THIS DOCUMENT	4
CORE CREDENTIAL LEARNING OUTCOMES	5
Agile Delivery Core	5
SPECIALIZED CREDENTIAL LEARNING OUTCOMES	10
Continuous Improvement	10
Improving Flow	12
SPECIAL THANKS	14

HOW TO READ THIS DOCUMENT

This document outlines the learning outcomes that accredited member organizations must address to offer ICA's Agile Delivery certification. To achieve the certification, a learner must complete the core credential plus at least two specialized credentials.

Each learning outcome (LO) follows a particular pattern, described below.

O.O Learning Outcome Name

Additional Context, describing why this Learning Outcome is important or what it is intended to impart.

The learning outcome's purpose further describes what is expected to be imparted to the learner (e.g., a key point, framework, model, approach, technique, or skill).

CORE CREDENTIAL LEARNING OUTCOMES

Agile Delivery Core

Agile Delivery is a 7–8 hour Core Credential for professionals who want to understand and participate effectively in modern agile delivery. It is framework–neutral and focuses on the common ideas and practices that sit underneath approaches such as Scrum, Kanban, XP, and hybrid ways of working.

Learners explore what makes agile delivery different; why customers and feedback sit at the heart of the approach; what makes a self–organizing agile team different and how it coordinates its work; how planning and estimation work in conditions of uncertainty; how delivery cadences, quality, integration, frequent delivery, and standards for when work is done support reliable delivery; and how teams respond to change without creating chaos.

1. What Is Agile?

1.1 Understanding Agile Delivery

Agile accepts that you can't predict complex work perfectly. Teams work in cycles, learn from real feedback, and adapt. It's a mindset, not a single prescribed process.

Build learners' understanding of agile as a fundamentally different approach to complex work. Introduce the mindset and core ideas behind agile delivery. Connect Agile values and principles with empirical and incremental delivery, frequent learning, collaboration, and adaptation so learners understand why agile delivery works differently from traditional task control.

1.2 Customer at the Heart of Delivery

Frequent customer feedback during delivery is how teams learn whether work creates customer value and adjust before too much is invested.

Develop learners' understanding of why customer and user involvement is fundamental to agile delivery. Explore identifying the customer or user, involving them throughout delivery, and creating regular feedback loops through reviews, demonstrations, conversations, observation, or other appropriate approaches. Help them recognize that "customer" isn't always obvious. The customer might be the end user, the person who paid for the work, the person affected by it, or all of them.

2. The Agile Team

2.1 What Makes an Agile Team Different

Agile teams shift from individual task ownership to shared responsibility for value delivery. This requires self-organization, close collaboration, and diverse skills within the team.

Build learners' understanding of the characteristics that distinguish an agile team. Explore shared ownership, self-organization, cross-functional collaboration, clear roles and responsibilities, and the shift from managing individuals and tasks toward enabling the team to deliver together.

2.2 Coordinating and Creating Shared Understanding

Self-organization does not mean people work independently without coordination. Agile teams need a shared understanding of goals, current work, decisions, blockers, and what others need from them so they can coordinate quickly without relying on a manager to direct every task.

Develop learners' understanding of the practices agile teams use to coordinate and maintain shared understanding. Help them see that coordination isn't just "communicate more" and that specific practices create different kinds of alignment. Explore techniques such as product backlogs, knowledge sharing, working agreements, visual information, daily coordination, collaborative conversations, and other techniques that help the team stay aligned as work changes. Explain the difference and impact of a practice done well that can create clarity or surface issues and a practice done poorly that can waste time or leave people confused.

3. Planning

3.1 Progressive Planning

Agile planning does not mean "no planning." Teams plan at different horizons with different levels of detail, and details get progressively clearer as they learn.

Build learners' understanding of progressive planning by contrasting it directly with traditional upfront planning. Teach how progressive planning is done in stages or horizons because in complex work details emerge. Teams make detailed plans for what they're doing now, rougher plans for what's next, and very rough plans for the far future. As they deliver and learn, they refine those rough plans with real information.

3.2 Breaking Down Work

Large work items create risk, take longer, and have more unknowns. Breaking work into smaller pieces lets teams deliver and validate incrementally, reducing risk and enabling faster feedback.

Build learners' ability to think about breaking work down into pieces that are independently valuable or testable. Help them understand that each piece of work can be completed by the team without waiting for other pieces, each piece can be validated, and the pieces together add up to the original goal. Introduce multiple techniques or strategies to decompose work into smaller pieces.

3.3 Estimation

Teams are better at comparing work sizes than predicting absolute time. Relative estimation techniques harness that strength and surface hidden complexity through discussion.

Develop learners' understanding of agile estimation and why teams estimate collaboratively. Relative estimation helps people discuss effort, complexity, uncertainty, or relative size. Cover the challenges that happen when an estimate is viewed as a firm, specific commitment. Introduce relative estimation techniques such as planning poker, t-shirt sizing, or affinity estimation.

4. Delivering

4.1 Agile Delivery Practices

Without regular practices, agile teams lose coordination and alignment. Understanding what each practice accomplishes helps teams implement them with purpose, not just habit.

Build learners' understanding of common agile delivery practices. Teach that these practices exist across frameworks but might have different names. The important thing is understanding why these practices matter. Explore how teams regularly plan or replenish work, coordinate day to day, review progress or outcomes, gather feedback, and reflect on how they work.

4.2 Quality & Integration

Frequent integration without quality creates chaos. Quality without integration hides problems. Together, they create speed that actually lasts.

Develop learners' understanding of building quality into delivery and integrating work early and often. Help them see the difference between teams that maintain quality and integrate continuously and teams that cut quality corners or integrate rarely. Cover how quality and regular integration work together such as teams that integrate frequently catch problems early (small, contained, easy to fix) and teams that maintain quality throughout can integrate confidently without fear.

4.3 Completion Clarity

Complex work never feels completely done. Teams need shared completion standards to set a boundary and know when to move forward.

Build learners' understanding of why shared standards for completion matter. Explore how a shared standard of done, such as Definition of Done, establishes what 'good enough to move on' means for a slice of work, protects quality and completeness, limits endless expansion, and makes it possible to finish an increment, gather meaningful feedback, and begin the next. Without a shared standard for what is enough, work can continue expanding and teams get frustrated, struggle to finish, learn, and move on. Help learners recognize that the standard for completion can vary by context and type of work.

4.4 Release Cadence

Release cadence isn't a prescription. Analyzing what factors matter helps teams choose the right cadence for their context.

Build learners' ability to examine what makes certain release cadences appropriate. Introduce key factors such as team delivery capabilities, organizational factors, feedback cycles, cost of delay, and customer needs or expectations. Help learners see that delivery cadence is about understanding constraints and trade-offs.

5. Responding to Change

5.1 Responding to Change Without Creating Chaos

Responding to change is central to Agile, but it does not mean changing direction every time a new request appears. Teams need to evaluate new information, decide what it means, update upcoming work when appropriate, and create reflection points to improve both the product and the way the team works.

Build learners' understanding that responding to change isn't reactive panic. It's deliberate adaptation based on new information. Introduce triggers for change such as customer feedback reveals different priorities, market conditions shift, technical constraints emerge, team capacity changes, or business goals evolve. Help learners distinguish between trivial changes (don't need adaptation), important changes (deserve thoughtful response), and fundamental shifts (require major adaptation).

SPECIALIZED CREDENTIAL LEARNING OUTCOMES

Continuous Improvement

Continuous Improvement is a 2–4 hour Specialized Credential for professionals who want to strengthen how teams learn from their work and deliberately improve their effectiveness. Agile teams do not improve simply because they hold retrospectives. Strong continuous improvement combines transparent observation, psychologically safe reflection, disciplined problem solving, small experiments, evidence, and follow-through. This credential focuses on retrospectives, Plan, Do, Check, Act / inspect-and-adapt thinking, root-cause analysis, improvement experiments, and mechanisms that turn lessons into sustained changes in how the team works.

1.1 Continuous Improvement as a Learning System

Teams often notice problems repeatedly without changing the conditions that produce them. Reflection becomes useful when observations are turned into a clear improvement target, a testable change, and evidence about whether the change helped.

Develop learners' understanding of continuous improvement as an empirical learning cycle. Connect transparency, inspection and adaptation with Plan Do Check Act / kaizen thinking, and distinguish ongoing improvement from occasional problem fixing or large transformation programs.

2.1 Retrospectives That Generate Learning

Retrospectives can become repetitive, superficial, or blame-oriented, producing long action lists that are quickly forgotten. Useful retrospectives create enough safety and evidence for the team to examine what happened, make sense of patterns, and select a small number of meaningful changes.

Develop learners' ability to design or participate in retrospectives that produce insight and action. Explore setting the stage, gathering data, surfacing different perspectives, identifying patterns, avoiding blame, prioritizing improvement, and closing with clear ownership.

3.1 Root Cause Thinking and Safe-to-Learn Experiments

Teams can jump from a symptom to a solution because the obvious fix feels faster. When the problem is poorly understood, the same issue often returns or the countermeasure creates new problems elsewhere.

Develop learners' ability to move from problem observation to disciplined improvement. Introduce techniques to discover underlying causes of problems such as problem framing, factual observation, root-cause techniques, Plan Do Check Act or A3-style thinking. Help learners understand what each technique does and when to use it. Introduce techniques for addressing the problem such as countermeasures and small safe-to-learn experiments.

4.1 Sustaining and Scaling What Works

Improvement actions frequently disappear because nobody owns the next step, the effect is never checked, or the team moves on before deciding whether the change should become part of normal practice.

Develop learners' ability to close the improvement loop. Explore ownership, visible improvement work, measures, review cadence, standardizing a successful change, adapting or abandoning an unsuccessful experiment, and escalating systemic obstacles the team cannot solve alone.

Improving Flow

Improving Flow is a 2–4 hour Specialized Credential for professionals who want to develop their ability to make work visible and improve the movement of work through a system. This credential focuses on work visualization, work in progress, blockers, dependencies, queues, policies, forecasting, and flow-oriented conversations that improve delivery outcomes.

1.1 Work Visualization and Flow

Knowledge work is often complex, making flow and progress difficult to detect and communicate. Visual systems create a shared understanding of how work is moving.

Build learners' understanding of visualization as a foundation for managing flow and collaboration. Visualization aids in making overload, delay, dependencies, and blocked work transparent.

2.1 Flow Constraints

Flow can be constrained by a variety of factors that are usually system conditions rather than individual performance problems.

Develop learners' ability to diagnose where flow slows and which systemic conditions contribute to delay. Constraints could include excess work in progress, queues, waiting, dependencies, batch size, rework, and context switching.

3.1 Flow Improvement

Flow improves through small, testable changes rather than broad process redesign. Teams need a clear constraint, a reversible intervention, and an early measure of whether the experiment is helping improve the flow of work to done.

Develop learners' ability to design practical, measurable experiments that improve flow without over-controlling work.

4. Ethical Use of Flow Measures

Flow metrics can support system improvement but become harmful when used as isolated data points or used to judge individual performance. Interpretation needs context and attention to what the measure actually represents.

Develop learners' ability to interpret flow measures constructively and avoid performance-management misuse. For example, a rising cycle time across a team reveals something real about the system, like growing work-in-progress or a bottleneck at a specific handoff, but using that same cycle time to rank individual engineers against each other misuses the measure.

SPECIAL THANKS

ICA would like to give special thanks to the contributors:
Kerri Sutey · Paulo Dias · Christina Hartikainen · Ahmed Sidky