SAFe Software Delivery and Agile Release Train Training Course

31 January – 4 February 2027
Nairobi

AGILE LEADERS
Training Center

AGILE LEADERS
Training Center

SAFe Software Delivery and Agile Release Train Training Course

Ref: 1708_228396 **Date:** 31 January – 4 February 2027 **Location:** Nairobi **Fees:** 4500 Euro

Overview

SAFe Software Delivery and Agile Release Train Training Course is a one-week course for delivery leaders, product teams, train engineers, Scrum Masters, architects, and software teams, who leave with a SAFe Delivery Operating Pack. The course connects Lean-Agile principles, Agile Release Trains, product and solution backlogs, PI Planning, architectural runway, flow, dependencies, DevOps, system demos, Inspect and Adapt, metrics, and implementation planning. Agile Leaders Training Center presents the course on SAFe software delivery and Agile Release Trains.

Who Should Attend

- Teams responsible for coordinating software delivery across multiple Agile teams
- Functions responsible for product vision, features, and backlog decisions
- Roles responsible for Agile Release Train events and impediment removal
- Architecture functions responsible for runway and technical alignment
- Teams responsible for DevOps flow, integration, and release readiness

The course assumes participants work with Agile software delivery and leaves out introductory Scrum, official certification preparation, portfolio funding design, and tool-specific configuration.

Departments and Industries

The course supports scaled software-delivery decisions across digital products, platforms, and technology-enabled operations.

- Product management and digital delivery departments
- Software engineering and quality teams
- Enterprise and solution architecture functions
- Financial services and telecommunications technology organizations
- Healthcare and public-service digital teams
- Manufacturing and mobility software operations

Learning Objectives

By the end of this course, participants will be able to:

- Apply SAFe Lean-Agile principles to delivery decisions
- Build Agile Release Train roles, events, and artifacts
- Prioritize features and enablers across connected backlogs
- Plan PI objectives, capacity, dependencies, and risks
- Analyze flow, DevOps, and system-demo evidence
- Evaluate improvements through Inspect and Adapt

Course Agenda

Day 1: SAFe Delivery Foundation

- SAFe Lean-Agile Principle Decision Map
- Development Value Stream Identification Canvas
- Agile Release Train Operating Model
- ART Role and Decision-Right Matrix
- Planning Interval Event and Artifact Calendar

Day 2: Product and Architecture Alignment

- Product Vision and Roadmap Alignment Board
- ART Backlog Feature and Enabler Structure
- Weighted Shortest Job First Prioritization Method
- Architectural Runway and Dependency Map
- Definition of Done and Built-In Quality Checklist

Day 3: PI Planning and Coordination

- PI Planning Readiness Checklist
- Capacity and Load Planning Worksheet
- Team Breakout and PI Objective Template
- ART Planning Board and Dependency Register
- Program Risk ROAM Decision Method

Day 4: Flow, DevOps, and Improvement

- ART Kanban and Work-in-Process Limit Design
- Flow Distribution, Velocity, Time, and Predictability Metrics
- Continuous Delivery Pipeline and DevOps Map
- System Demo Evidence and Feedback Record
- Inspect and Adapt Problem-Solving Workflow

AGILE LEADERS
Training Center

Day 5: SAFe Delivery Practice

- Value Stream and ART Design Exercise
- Backlog and Architecture Alignment Exercise
- PI Planning and Dependency Exercise
- Flow Metric and Inspect and Adapt Exercise
- SAFe Delivery Operating Pack Capstone Exercise

Practical Exercises

The course uses suggested activities to practice SAFe software-delivery decisions.

- Suggested activity: map a development value stream, ART roles, decision rights, and event cadence for a software product.
- Suggested activity: prioritize features and enablers, then connect the product roadmap with architectural runway and quality criteria.
- Suggested activity: prepare PI objectives, capacity, dependencies, risks, and an ART planning board.
- Suggested activity: review flow and system-demo evidence and assemble improvement actions in a SAFe Delivery Operating Pack.

FAQs

Who suits SAFe software delivery training and what does it assume?

SAFe software delivery training suits product, delivery, train, architecture, engineering, quality, and DevOps professionals. It assumes practical exposure to Agile team delivery but not prior responsibility for an Agile Release Train.

How does SAFe software delivery training differ from Scrum training?

SAFe software delivery training coordinates multiple teams through value streams, Agile Release Trains, PI Planning, shared backlogs, architecture, dependencies, system demos, and improvement. Scrum training focuses primarily on one team's roles, events, and artifacts.

What does an Agile Release Train coordinate in SAFe software delivery?

An Agile Release Train coordinates product direction, team plans, features, enablers, architecture, dependencies, integration, demonstrations, flow, risks, and improvement across teams working on a shared solution.

What are the main outputs of PI Planning in SAFe software delivery?

PI Planning produces aligned team plans, PI objectives, capacity decisions, dependency visibility, an ART planning board, identified risks, and a shared view of planned value delivery.

AGILE LEADERS
Training Center

How does Inspect and Adapt improve a SAFe Agile Release Train?

Inspect and Adapt improves a SAFe Agile Release Train by combining a system demo, measurement review, retrospective evidence, structured problem solving, and improvement backlog items for the next planning cycle.

Conclusion

Participants take a SAFe Delivery Operating Pack back to work. It connects value-stream design, ART roles, backlogs, architecture, PI objectives, dependencies, risks, flow metrics, DevOps, system demos, and improvement actions. It changes separate team plans into coordinated delivery decisions, visible cross-team commitments, measured flow, and traceable improvement work.