

SYLLABUS

AGILE MASTER CERTIFICATION

Lesson 1

- 1.0 Class objectives and Overview
- 1.1 Course Introduction
- 1.3 Super Agile: A New Way of thinking
- 1.3 Agility and Superperformance
- 1.4 Agile and Operations Excellence
 - 1.4.1 PDCA and Scrum
- 1.6 Agile Manifesto
- 1.7 Why Super Agile: Advantages
- 1.9 Value Optimization and Delivery
- 1.9 True North of Superperformance
- 2.0 Acceleration of Enterprise Agile

Lesson 2

- 2.0 Class Objectives
- 2.1 What makes Scrum Adoptions Succeed
- 2.2 Super Team
 - 2.2.1 Small (less than 10)
 - 2.2.2 Self-organized (no formal roles)
 - 2.2.3 Empowered and Enabled (Tools and Support)
 - 2.2.4 Collaborative (co-located or virtual teaming)
- 2.3 Super Coaching
 - 2.3.1 Servant Leadership

2.3.3 Super Coaching

2.3.3. Facilitation and Conflict Resolution

2.4 Super Team Growth

2.4.1 Forming

2.4.2 Informing

2.4.3 Transforming

2.4.4 Superperforming

Lesson 3

3.0. Class Objectives

3.1 Adopting Super Agile

3.2 Patterns for planning a transition to Super Agile

3.3 Skills Adoption Continuum

3.4 Command Control and Super Agile

3.5 Towards self-organization and emergence

3.6 Organizational, Physical and cultural changes needed

Lesson 4

4.0 Course Objectives

4.1. Other Agile frameworks

4.1.1 Different development methodologies

4.1.2 Scrum

4.1.3 XP (Extreme Programming)

4.1.4 Lean/Kanban

4.1.5 SAFe (Scaled Agile)

4.2 Linking Agile to Dev Ops: Higher Value through Faster Deployment

4.3 Governance and Agile

4.4 Scrum events and artifacts

4.4.1 Agile Values

4.4.2 Iteration and Spring Lifecycle

4.4.3 Key Agile events

4.4.4 Planning and prioritization

4.4.5 Demos

4.4.6 Retrospective

4.4.7 Daily standup meetings

4.4.8 Product backlog

4.4.9 Sprint backlog

4.4.10 Deliverables

4.4.11 Definition of Done

Lesson 5

5.0 Course Objectives

5.1 Key Agile Rolew

5.1.1 Sponsor, Scrum Master, Product Owner and Team roles

5.1.2 What they should and should not do

5.1.3 Attributes of good Scrum Masters or Product Owners

5.1.4 Failure modes

5.1.5 Scrum Team member role

5.2 Agile estimating, planning and monitoring

5.3 User Stories and Epics

- 5.4 Managing Non-Functional Requirements
- 5.5 Writing “good” stories
- 5.6 Basics of prioritization
 - 5.6.1 Value based prioritization
 - 5.6.2 Kano model
 - 5.6.3 Karl Wieggers Relative Weight Method
- 5.7 Velocity: How to find it and how to use it?
- 5.8 Release or roadmap planning
- 5.9 Units of Estimation: Story points and ideal time
- 5.10 Techniques of estimation: Planning poker and Affinity estimation
- 5.11 Methods of Tracking Agile projects: Burndown charts, Kanban boards and other indicators

Lesson 6

- 6.0 Class Objectives
- 6.1 Using Super Agile on Complex projects
 - 6.1.1 Scaling agile at enterprise level
 - 6.1.2 Scaling Product Owners and Scrum Master Role
 - 6.1.3 Scaling the Backlog
 - 6.1.4 Coordinating multiple scrum teams
- 6.2 Scaled Agile Foundation (SAFe®) introduction
- 6.3 Agile in distributed teams
- 6.4 When to apply agile (and when not to)
- 6.5 Tools for agile project management
- 6.6 Agile Testing Pyramid
- 6.7 What Comes Next (Practicum)
 - 6.7.1 Fieldwork in Slack
 - 6.7.2 Coaching Practice with Individuals and Groups

6.7.3 Webinars and Conference Calls and Meetups

6.7.4 Community of Practice Development