Agile Education Checklist

Category
Agile Process

Purpose
The Agile Education Checklist is a reference framework to use when determining what content should be included in an Agile education program. This framework allows organizations and teams to explore what educational topics are needed to support the adoption of Agile practices and starts the discussion on how Agile practices will fit into the organization. When complete, participants will have an initial backlog of content for their education program.

Description

Framework name:	**AGILE EDUCATION CHECKLIST**
Participants:	- Management and senior leaders of an organization looking to apply Agile practices. - Member of the HR department and/or someone responsible for corporate learning and development of the staff. - A knowledgeable, experienced Agile practitioner as the facilitator
Framework goal:	To create an initial backlog of content for a customized Agile education program and identify the participants who will attend the training.
Framework guidelines:	- Define a clear and measurable goal, or vision, for what you wish to achieve with the Agile education program. OKR or SMART goals are two frameworks to consider when crafting your objective. - Develop a broad list of participants for the organization's training program. Consider people who directly apply practices, i.e., Scrum Teams, and other groups such as HR, markets, DevOps, etc.. - Select the Agile framework that will be used as the foundation for your education program: Scrum and/or Kanban. - Select the additional topics and\or practice areas that will be included in the program. - Identify the relevant organizational contexts that may influence your Agile adoption and will need to be communicated to the participants. Some areas to consider might include roles, security, compliance, governance, etc.
Boundaries provided:	Total duration of the program should not exceed sixteen total hours for courses in the curriculum.

	• Content should be foundational and applicable to all audience members. Follow-up sessions to discuss role specific concerns can be identified later, perhaps through learner feedback.
Resources used:	• In-person: Whiteboard or wall, post-it note, syllabus, Agile education checklist • Online: Miro or similar online collaboration tool, syllabus, Agile education checklist
Actionable outcome(s):	1. A list of participants to invite. 2. A high-level set of topics to be included in an Agile education program.

Source Notes

The Agile Education Checklist was created by Joel Bancroft-Connors.

Agile Education Checklist

Objective:		
Agile Foundation:	☐ Scrum	☐ Kanban
Specific Practices:	☐ Estimation	☐ Vertical Story Slicing
	☐ User Story Writing	☐ Backlog Refinement
	☐ Prioritization	☐ Sprint Review
	☐ Managing WIP	☐ Principles for Scaling
	☐ Metrics	☐ Introduction to Technical Practices
	☐ Personas	☐ Definition of Done
	☐ _____	☐ _____
Total Duration:		
Organizational Requirements:		

APPLIED FRAMEWORKS

Explanation of Key Terms

Agile Foundation: which Agile framework will you use as the foundation for your education program: Scrum, or Kanban.

Framework	Type of Work	Duration
Scrum	We recommend selecting Scrum when the work is primarily focused on feature-driven outcomes, the end goal is not well understood and\or it is not exactly clear how to get to the final objective. New product development often works best using Scrum.	~6 hours
Kanban	We recommend selecting Kanban when the work is generally small pieces of work such as defect fixes, or small enhancement, on an existing product. Kanban often works best for activities and teams when a fast response time is crucial to satisfy the customer.	~4 hours

Specific Practices: are the supplemental topics which address specific challenges the team are experiencing today or are anticipated in their new way of working. The table below provides further explanation for the topics in the checklist based on our experience working with our clients. You may find your teams face other challenges. If so, work with your facilitator to identify the problem and a potential solution that can be worked into your curriculum.

Topic Area	Common Symptoms	Duration
Estimation	● Is the team estimating in physical time, hours, days, etc.? ● Do they have trouble completing work within a given timebox?	1 hour
Vertical Story Slicing	● Are you having difficulty delivering end-user value in four weeks or less? ● Is your work technical in nature and several items are needed to deliver value to the customer?	1 hour
User Story Writing	● Can a clear "What" and "Why" be expressed for each requirement? ● Are requirements written as discrete pieces of value from the user, or customer, perspective?	2 hours
Personas	● Is there clarity on who the customer, or end user, is? ● Can you clearly answer if the customer is receiving value?	45 minutes
Backlog Refinement	● Does the team have enough work to keep them busy for four to six weeks? ● When work starts, does it frequently need to stop again to gain additional clarity?	30 - 45 minutes
Definition of Done	● Does finished work require re-work or bug fixing after it is complete? ● Is work still in-process, even if team members say they are	30 - 45 minutes

APPLIED FRAMEWORKS

	"finished"?	
Prioritization	• Is everything Priority One, High or Urgent? • Do priorities change based on who you are talking to? • Can you clearly connect your strategic initiatives to the work being done by the teams?	2 hours
Sprint Review	• How often will the work be reviewed by stakeholders? • Will multiple teams be engaging with the same stakeholders?	30 minutes
Metrics	• Is the organization using Velocity as a measure of output? • Can the team use past results to predict future results? • Can the team measure its health and improvement over time?	30-45 minutes
Managing WIP	• Does the team have more unfinished work than finished work at the end of their iterations? • Is the cycle time to complete work high?	1 hour
Principles for Scaling[1]	• Are three, or more teams, working on the same work product simultaneously? • Are teams highly dependent on one another to complete their work? • Do integration issues between teams lead to long cycles of rework or quality issues?	1-2 Hours
Introduction to Technical Practices[2]	• Is quality an ongoing issue? • Is the team spending more and more time in completing work every iteration due to technical overhead?	45 min

---

[1] This course would provide an introduction to the scaling principles and would enable multiple teams to work together with less confusion and thrashing. It is not a replacement for a comprehensive workshop on which scaling approach would be best for the organization.

[2] This introduction highlights the importance of key technical practices for developing software-powered products and how they can be leveraged to improve the ability to release working increments of work on a regular cadence. It is not a replacement for a comprehensive, hands-on workshop to improve technical practices.

APPLIED
FRAMEWORKS