Build in Quality at Scale with
Acceptance Test-Driven Development / Behavior Driven Development

"Built-in Quality" is a core value of SAFe®. It allows quicker delivery of business value. One of the key practices in realizing built-In quality is Acceptance Test-Driven Development (ATDD). This course describes how the triad (customer, tester, and developer) creates acceptance tests to provide a joint understanding of the requirements. It shows how to use those acceptance tests as a communication and verification tool. Applying these skills streamlines communication within the organization, decreasing rework, raising customer satisfaction, and promoting trust within the organization. These methods have demonstrated an ability to be able to lower released errors by up to 90%. This course is based on the SAFe recommended book *Lean-Agile Acceptance Test-Driven Developmen*t by Ken Pugh.

SAFe's DESCRIPTION OF ATDD

> Built-in Quality is one of the four core values of the Scaled Agile Framework®. The enterprise's ability to deliver new functionality with the fastest sustainable lead time and to be able to react to rapidly changing business environments is dependent on solution quality.

> But built-in quality is not unique to SAFe. Rather, it is a core principle of the Lean-Agile Mindset, where it helps avoid the cost of delays associated with recall, rework, and defect fixing. The Agile Manifesto is focused on quality as well; it says, "Continuous attention to technical excellence and good design enhances agility." There can be no ambiguity about the value of built-in quality in large-scale systems. It is mandatory.

> The goal is to have the system work as intended, not to simply have the code do as intended. Lean-Agile thinking achieves this is by moving testing forward in the development process.

> Acceptance Test-Driven Development (ATDD) applies this test-first approach to testing stories and features and capabilities. Using ATDD, teams write the test first before developing the code. Whether ATDD is adopted formally or informally, many teams find that this approach is much more efficient.

> In his book, *Lean-Agile Acceptance Test-Driven Development*, Ken Pugh notes that the focus of ATDD is more on expressing requirements in unambiguous terms rather than focusing on the test per se.

> The goal is for the customer, developers, and testers to collaborate and get a clear understanding of requirements prior to implementation.

> Pugh notes that there are three names that are used for this approach: Acceptance Test-Driven Development, Specification by Example, and Behavior-Driven Design. While there are slight differences in these three versions, they all focus on the same goal.

> Whether it is viewed as a form of expressing requirements or as a test, it results in better understanding. Acceptance tests serve to record the decisions made while writing stories.

SAFe References to Built-in Quality and Test-First
http://www.scaledagileframework.com/built-in-quality/
http://www.scaledagileframework.com/test-first/

OBJECTIVES

This course helps participants understand how to transform requirements accurately into testable specifications. This is a collaborative, efficient manner that minimizes waste. It addresses requirements, specifications, implementation and testing. You will learn:

- Principles of Acceptance Test-Driven Development
- How to turn requirements into acceptance tests
- What is a good acceptance test
- How to use acceptance tests as a communication vehicle
- How ATDD embodies SAFe's core principle: Build Quality In

INSTRUCTOR

Ken Pugh, author of *Lean-Agile Acceptance Test-Driven Development: Better Software Through Collaboration*.

TARGET AUDIENCE

Customers, product managers, business analysts, SMEs, developers and testers.

It is essential that the development and test team and at least one customer, business analyst, product manager, or SME attend the workshop together.

This course is appropriate for anyone who is involved in the definition, development and quality assurance of software related products.

OUTLINE

- Software Development
    - What is ATDD
    - Why ATDD is useful
    - The process
    - The roles and responsibilities
- Acceptance Test Examples
    - Acceptance test style, size, scope, clarity
- The Business Tests
    - Objectives
    - Scope
- User Stories and Scenarios
- Test Anatomy
- Tables as Tests
- System Boundary and Tests
- Events, Responses, States
- Complex Business Rule and Separation
- Test Evaluation
    - Common pitfalls and how to avoid them
    - Maintainability, scalability

- o   Sustainability
- Retrospective
  - o   Transition Issues
  - o   Motivation issues

ATTENDEE MATERIALS

Workshop materials are provided at the start of the class.

ROOM SETUP AND EQUIPMENT

Students usually sit at tables, 4-6 students per table

Flip chart and whiteboard for the instructor

A projector with screen

PREREQUISITES

Knowledge of SAFe

COURSE LENGTH

This course is structured with a workshop for all participants to present common concepts followed by individual team workshops to cover team-specific issues and to develop acceptance tests for stories.

- 1 1/2 days: All teams together
- 1 1/2 day: Teams individually (1/2 day per each)

MAXIMUM

Three teams

NOTES

Some of this description is reproduced with permission from Scaled Agile, Inc. Copyright © 2011-2017 Scaled Agile, Inc. All rights reserved.   SAFe and Scaled Agile Framework are registered trademarks of Scaled Agile, Inc.

This course does not provide any certification related to Scaled Agile, Inc. or SAFe.

PROVIDER

Ken Pugh, Inc. 732 Ninth Street #695, Durham, NC 27705.

Course copyright © 2007-2017 Ken Pugh

CONTACT

Leslie Killeen 919-490-6335

leslie@kenpugh.com