

Hands-on TDD Workshop

Elevate your coding game! This intensive 2-day Test Driven Development (TDD) workshop will introduce you to the skills you need to improve your coding experience, productivity, and flow. Learn how to achieve seamless collaboration with your Product Owner and testers. Master advanced TDD techniques, from property-based testing to legacy code management. Our expert-led guidance and interactive exercises empower you to craft high-quality code faster, while championing a quality-first mindset. For those seeking an even deeper dive, join our optional third day for advanced exercises and topics. Take your skills to the next level and lead the charge in smarter, more effective value creation.

Objectives:

- ✓ Understand the purpose of TDD and how it enhances team agility
- ✓ Learn how to integrate TDD into the overall workflow of your team and improve collaboration between product owner, developer, and tester
- ✓ Get hands on experience with the TDD cycle
- ✓ Understand how to use TDD to understand and extend existing code bases
- ✓ Learn the proper use of test doubles and property based testing
- ✓ Learn how TDD integrates with and enhances your practice of CI/CD

Audience:

While this hands-on course is primarily intended for developers, product owners and testers with at least some technical background will benefit from understanding how to leverage TDD to improve team collaboration.

Day 1: Understanding Test Driven Development

Introduction to TDD in an Agile Workflow (45 mins)

The Foundations of TDD
TDD's Integral Role within Agile Practices
Benefits and Principles for Accelerated Development

TDD Process Overview (45 mins)

The Red-Green-Refactor Cycle Demystified
Unveiling the Power of Iterative Development

Improving Developer Experience and Team Effectiveness (30 mins)

Harnessing TDD's Immediate Benefits
Swift Feedback for Confident Coding
Fostering Collaboration and Shared Understanding
Seamless Integration with CI/CD Pipelines
Big Up-Front Design vs. Emergent Design
Elevating Code Quality Through Iteration

Your First TDD Session: Code Along with the Instructor (90 mins)

Dive into Practical TDD Experience
Setting Up the Coding Environment
Looking at a User Story with TDD in Mind
Our First Red-Green-Refactor Cycle
Embrace "Coding by Intention" Approach
Evolve the Design with each Red-Green-Refactor Cycle
More Tests for Comprehensive Coverage
Review and Q&A

Hands-On Exercise 1: Basic TDD Techniques (Part 1) (90 mins)

Our Class Domain, Feature, and Stories
Team Up for Pair Programming and TDD Exploration
Achieve Checkpoint 1: Make the First Acceptance Criteria Pass
Pair Solutions Presentation and Review
Discussion: Applying What You've Learned on Your Team

Hands-On Exercise 2: Basic TDD Techniques (Part 2) (90 mins)

Improve Your Collaboration with POs and Testers Through TDD
Discover missing tests by asking "what if...?"
Iterate Your Solution and its Design with More Tests
Checkpoint 2: Final Solution for Acceptance Criteria 1
Final Pair Solution Presentations and Review
Discussion: Applying What You've Learned to Your Team

Day 2: Intermediate TDD Techniques and Integration

Day 1 Review and Q&A

Introduction to Test Doubles (Mocks, Stubs, Fakes) (45 mins)

Why use test doubles in TDD?
Types of test doubles
Maintaining isolation between components during testing

Hands-On Exercise 3: Incorporating Test Doubles (90 mins)

Our Next Acceptance Criteria
Interfacing with an Existing Component
Writing or Using Existing Tests to Understand an Interface
Writing a First Test with a Test Double
Complete the Test, Code, and Test Double Iteratively.
Checkpoint 3: Acceptance Criteria 2 Passing
Pair Solution Presentation and Review

Hands-On Exercise 4: Using TDD to Drive Continuous Integration (120 mins)

Our Next User Story
Working from a Common Code Base
Agreeing on a Basic Design
Working on a Story In Parallel
Integrating Frequently and Pull before Push
Resolving Conflicts and Broken Tests
Checkpoint 4: Working Solution
Solution Presentation and Group Discussion

Group Discussion: Applying What You've Learned on Your Team (45 mins)

Impacts on Team Collaboration and Flow
Implications for Code Quality
Effect on Design Quality

Day 3: Advanced TDD Techniques and Best Practices (Optional)

Day 2 Review and Q&A

Hands-On Exercise 4: Complex Scenario and Refactoring (180 mins)

Extending an Existing Code Base
Understanding the Code Through its Tests
Confirming Understanding by Writing Missing Tests
Agreeing on a Design Approach
Implementing a New Story as a Team
Evolving the Design Iteratively
Using Test Doubles to Isolate the Code to be Changed

Continuous Integration and Continuous Delivery with TDD (45 mins)

How TDD contributes to reliable CI/CD pipelines
Automated testing and deployment using TDD-based tests
Ensuring confidence in code changes through automated testing

Advanced TDD Techniques and Topics (45 mins)

Property-based testing for comprehensive test coverage
Handling edge cases and corner scenarios
Strategies for dealing with legacy code using TDD
Effective Test Codebase Management
Tracking the Value of TDD with DORA Metrics

Wrap-Up and Future Learning (30 mins)

Class Summary
Resources for Continued Learning



www.IconAgility.com

(636) 530-7776

Final Q&A and Feedback (30 mins)

Final Questions and Reflections on the Class

Recommendations to Improve the Class Content, Flow, and Structure