

Course: Scrum Developer

ID: 1384-SCRMD

CDUs/Contact Hrs / PDU: 21

Course Length: 3 Days

Course Description: This three-day intensive and highly interactive course focuses on the processes, tools and techniques of Scrum from the perspective of a Scrum Developer. This Course employs practical exercises, case studies and discussions to provide everything that a Scrum Developer needs to know to be able to deliver feature rich, valuable software.

Course Objectives:

- Learn agile engineering best practices
- Learn to write software which fully supports an iterative process without extensive rework
- Learn to implement design alternatives and avoid overdesign while practicing Just-in-Time development
- Use Object Oriented techniques to develop truly extensible applications
- Learn the keys to collaboration, paired programming, and cross functional teams
- Learn to Support collaborative code ownership and embrace a common aesthetic

Target Audience: This course will benefit architects; business analysts, developers; QA testers and engineers, and project/product managers.

Prerequisites: A basic understanding of Scrum

Provided Material:

- Course Binder
- Pro Agile .NET Development with Scrum (Book)

Course Outline:

- **Lesson 1:** Collaboration
 - How Agile teams work together
 - The concept & reality of one team
 - The customer as a team member
 - Paired programming
 - Collaboration exercise
- **Lesson 2:** Architecture & Design
 - Basic architecture & design
 - Enabling testability
 - Ensuring refactoring ease
 - Principles of architecture in an agile environment
 - Design practices on an Agile team
 - Paired programming exercise
- **Lesson 3:** Test driven development (TDD)
 - TDD as a design approach
 - The red-green-refactor cycles
 - Unit testing principles and practices
 - What makes good tests?
 - Measuring test effectiveness
 - TDD exercise
- **Lesson 4:** Refactoring
 - A basic introduction in refactoring
 - When to refactor
 - Refactoring to patterns
 - Refactoring for maintainability
 - Practical applications of refactoring
 - When NOT to refactor
 - Refactor exercise
- **Lesson 5:** Continuous Integration
 - Key practices of continuous integration
 - Single Command build
 - Creating automated builds
 - Self-testing
 - The concept of fast
 - Using a single source repository
 - Increasing visibility and automating deployment
 - The continuous integration exercise
- **Lesson 6:** Implementation frameworks & retrospectives.
 - How to implement course concepts in the real world.



Looking Glass Development, LLC

P.O. Box 630516 Littleton, Colorado 80163-0516 303 663.5402 888 338.7447
www.lookingglassdev.com information@lookingglassdev.com