This 2-day course assures students understand what adopting Scrum will mean for their organization and themselves, and to make passing Certified Scrum Master training a certainty. The course begins with the concepts and terminology of iterative development: developing and delivering portions of a total product according to a well-defined schedule and partitioning of product features. The business case for iterative development is thoroughly covered. The course then discusses the principles and practices that define an agile approach to software development, including: delivering continual value to the customer, flexible and rapid response to change, short time-boxed iterations, and rapid feedback on project state. The course next covers each of Scrum’s practices and, most importantly, the structure and flow of how a Scrum project is conducted according to agile principles. Extensive exercises allow students to plan a release, estimate user stories and tasks, plan and populate a sprint, and understand how to conduct and end a sprint, with special consideration of software deployment options.

**Audience:** Individuals who need to understand Scrum, agile development in general, and the relationship between iterative development and agile development.

**Prerequisites:** Experience in software development, project management, or business or systems analysis is desirable, but not mandatory.

**Number of Days:** 2 days

---

1. **Iterative Development**
   - The Iterative Philosophy
   - Structure of a Typical Iteration
   - The Business Case for Iteration

2. **Agile Development**
   - Agility – What Does it Mean?
   - The Agile Manifesto
   - The 12 Agile Principles
   - Agile Practices

3. **Scrum**
   - Scrum Practices
   - Structure of Scrum
   - 3 Work Products
   - 3 Project Roles
   - 4 Project Meetings

4. **User Stories & Requirements**
   - What is a User Story?
   - What Does a User Story Look Like?
   - Where Do User Stories Fit in Scrum?

5. **Planning a Scrum Project**
   - The Product Backlog
   - Mapping Features to Product Backlog
   - Identify User Stories from Features

6. **Agile Estimation**
   - Story Points & Ideal Days
   - Estimating Actual Effort
   - Velocity
   - Velocity & Actual Time
   - Estimating with Planning Poker

7. **Planning a Scrum Sprint**
   - Mapping a Sprint Backlog to Tasks
   - The Spring Planning Meetings
   - Velocity-driven Planning
   - Commitment-driven Planning

8. **Executing a Sprint**
   - The Task Board
   - The Daily Scrum
   - Accumulating the Burndown
   - Team Self-Management
   - Aborting a Sprint
   - Finishing Early or Late
   - Testing with the Sprint
   - Bugs in an Iteration
   - Ending the Sprint
   - Deploying the Software
9. Scrum’s Effect on Stakeholders
   Business Analysts
   Developers
   Project Managers
   Testers
   Documentation Writers

10. Scaling Scrum
    Planning for Dependencies
    Planning for Multiple-Team Projects

11. Appendix A – Agile Alternatives
    Extreme Programming
    Agile Unified Process