Applying Scrum with Use Cases

Leveraging the use case requirements approach and Scrum’s agile project management approach gives you the ability to quickly develop working software that meets business needs. In Scrum, teams work in 30 day sprints to produce production-quality software driven by business priorities and using an empirical, adaptive approach. This course explains this approach’s values, key techniques, and how each team member contributes to success. This course blends ideas from Scrum, Extreme Programming (XP), and the Rational Unified Process (RUP).

You will learn how to build a product backlog using a use case model as input, and how to plan and work in 30 day sprints. You will apply what you learn in a series of exercises that take you through project start-up, sprint planning, and running a sprint.

Objectives:
- Understand key concepts and terms
- Be able to explain the methodology’s benefits and key techniques
- Become familiar with the progression of key activities and artifacts
- Prepare to participate in an upcoming project

Audience:
This course is intended for all who will actively participate in a project using this approach. People on the edges of the project (secondary customers, management staff, and participants in related processes) will also benefit.

Pre-requisites:
IconATG’s “Defining & Managing Requirements with Use Cases” or a basic understanding of use case models and use case specifications.

Duration:
2 days (Scrum / Agile workshop days available at a discounted rate)
Outline:

1. Agile Methodology Fundamentals
   - What are agile methods?
   - What is iterative development? Covers the basics and principles of iterative development and industry accepted best practices.
   - Agile methods’ values
   - Intro to Scrum/benefits vs. waterfall Brief history of Scrum, comparison to waterfall, advantages of Scrum.
   - Exercise: Characterize Your Current Process. Students will break into small discussion groups to discuss their current process and what challenges or obstacles they may face when adopting agility

2. Scrum in a Nutshell
   - What is Scrum and an overview of Scrum process flow
   - Scrum Roles and responsibilities – The ScrumMaster, project team. Covers the roles played within a project team, with an emphasis on the ScrumMaster.
   - How does Scrum work and how do we conduct project planning?
   - Exercise: Understanding Scrum Roles. Students will break into small discussion groups to determine how to use scrum to bring a project back on track.

3. Traditional Management vs. Scrum
   - Agile Leadership Principles – Servant Leadership, Empirical Management, Empowerment, Quality-First, Continuous Improvement, and “Standardisation”

4. Use Case Overview
   - What are use cases?
   - Describe how to develop a use case model
   - Describe the basic process for writing a use case specification and describe the components of a use case specification
   - Discuss the relationship between use case and user interfaces
   - Review tips for writing quality use case specification
   - Exercise: Developing a Use Case Specification. Students will have the opportunity to write a use case specification (optional exercise)

5. Planning and Estimating
   - Planning Overview. Why do we plan and what makes a good plan?
   - How is agile planning different? Planning takes place at the Day, Iteration, and Release level
   - Estimation of Size and Duration. Story points vs. ideal days. We will then extrapolate the story point concept and explain how this applies to the concept of use case points.
   - Exercise: Estimating for the Product Backlog. Students will work in teams to estimate items in the product backlog using points.
   - Ideal days. A discussion about Ideal Time vs. Elapsed Time
   - Planning Poker. Introduces the class to planning poker and how to conduct a planning poker sessions.
   - Exercise: Planning Poker. Students will work in teams and use planning
poker as a means to estimate items in a backlog.

6. Project Initiation and Release Planning
   • Role of the product owner
     Lays out the responsibilities and expectations of the product owner and his/her relationship to the project team.
   • Establishment of project charter
     Overview of basis for scope of work, starting point for product backlog. Describe other project initiation activities.
   • Product backlog
     Definition of the product backlog, how it is established. Demonstrate how to size use cases.
   • Exercise: Build a product backlog
     Allow students to go through the steps of establishing the product backlog and size use cases.

   • Setting priorities
     How the product backlog gets prioritized and maintained. Focus is on the role of the product owner.
   • Release planning
     Demonstrate how to use prioritized backlog, estimated velocity, and sprint goals to establish a release plan.

7. Planning a Sprint
   • Conducting Sprint Planning Meeting
     Part 1: Selecting a subset of the product backlog
     Discuss how to determine the sprint scope including hangover.
     Part 2: Developing the sprint backlog
     Discuss how to avoid hangover, estimate velocity, and establish a sprint goal.
     Exercise: Build a sprint backlog
     Students will be provided a product backlog and preliminary release plan, to build a sprint backlog.
   • Developing sprint tasks
     Topics covered include task granularity, dependencies, scheduling and assignment.
   • Exercise: Establish sprint plan tasks
     Students will create and plan iteration tasks.

8. Running a Sprint
   • Conducting the daily Scrum
     Discuss the focus and conduct of the daily Scrum. Learn what is included and what is excluded. Learn the value of the daily Scrum.
   • Managing the sprint scope
     Discuss tips and techniques for insulating the team from distractions. Learn how to handle the sprint being behind or ahead.
   • Warning signs and how to address them
     Covers common pitfalls and avoidance strategies and recipes for failure.
   • Tracking progress/burndown
     Methods for managing task completion. Covers burndown and progress tracking.
   • Exercise: Build and maintain burndown (59 minute Scrum)
     Students break into teams and simulate an actual sprint. They plan the sprint, create tasks, status the tasks, and hold daily Scrums all within 59 minutes.

9. Closing a Sprint
   • Conducting a sprint review
     Demonstrate what is expected and covered in a sprint review.
   • Conducting a sprint retrospective
     Discuss the purpose and scope of a sprint retrospective.
• Inspecting and adapting the process
  Discuss a key aspect of Scrum: inspect and adapt. Emphasizes the importance of self-directed teams.
• Reporting project status
  Discuss metrics, hangover, and velocity.
• Updating the backlog
  Covers re-prioritization, injects and budget constraints.
• Closing the project
  Summarization and definition of done.