

Agile Development Best Practices Course Outline

1. Introduction

- A team can't be agile if its organization won't allow it. This section provides a short overview of Agility and the challenges one faces in both High Ceremony and undisciplined environments.
- Agile Software Overview
- Team Values
- Supporting Environment

2. Agile Management

- The properties of agile management processes are discussed, and a number of agile practices are explored. The practices discussed show how an agile development can be managed while simultaneously providing the visibility necessary for management and the maneuverability that developers need. This is the heart of the course, and has an embedded role-playing exercise.
- Natural Rhythms of Development
 - Product Lifecycle
 - Release
 - Iteration
 - Day
- Release Activities/Practices
- Iteration Activities/Practices
- Daily Activities/Practices
- What about Maintenance

3. The Technical Processes

- In order to be agile the team must be able to produce *Quality Requirements* and *Quality Code* that can be changed, managed, and prioritized in an agile fashion. In this section we discuss ways to do this. We introduce a Use Case Driven approach to requirements called the Ever Unfolding Story to meet the requirements need, and a number of technical practices that help produce *Quality Code* from these requirements. This section provides merely an overview of these practices, leaving their details to other courses.
- Driving a project with Use Cases
- Decomposing Use Cases into software requirements - the Ever Unfolding Story
- Properties of Quality Code
- Unit Testing
- Test Driven Development
- Refactoring
- Pattern Oriented Design