SUMMARY

This workshop equips the software triad (customer/Business stakeholder, developer, and tester) turn requirements into software-development specifications. They learn how to use Acceptance Test-Driven Development (ATDD) using Behavior-Driven Development (BDD) to create a non-controversial definition of done. It almost eliminates the chance of misunderstandings. By creating this agreement on the Definition of Done before starting the development and testing efforts, many of the dilemmas and resulting problems listed above are prevented outright. It uses a combination of training and 1-1 coaching. There is also the opportunity for follow-on coaching.

DESCRIPTION

AGILE TEAMS FACE MANY CHALLENGES

Agile teams face multiple dilemmas when it comes to requirements. Requirements can be

- Ambiguous and controversial in that not everybody agrees on what really should be delivered or what the outcome should be
- Vague and don’t convey enough information for a team to provide an accurate and predictable estimate of the effort to implement it
- Incomplete in that they do not represent all the necessary stakeholder expectations
- Tied to a specific solution rather than describing the actual problem
- Transient in nature, such as when User Stories or other types of Backlog Items (representing new requirements) are marked as Done and then archived, and do not represent a persistent, holistic, and clear set of specifications that remain available, and are maintained to be representative of the current state of the product at all times
- Too technical for Product Managers, Product Owners, and other Business stakeholders to understand or support, such as technical debt or architectural requirements
-Disconnected from, and hard to connect back to, larger initiatives, especially when many groups and teams are contributing to these initiatives
- Difficult to split into smaller requirements that can be fully completed within teams’ planning horizons, such as a two-week Sprint, while still delivering “value”

These dilemmas often lead to issues such as

- Poor product quality
- Slipped delivery or commitment dates
- Unmet expectations
- Dissatisfied stakeholders and customers
- Lack of trust
- Long delays;
- Loss of enterprise knowledge
- Unnecessarily high development and support costs

The root cause problem of these is that so-called Agile requirements (such as epics, features-with-benefits, and user stories) are not software specifications. They are value statements. Software developers and testers must do their best to interpret these value statements and produce something that satisfies the expectations of the customer or business stakeholder; however, regardless of the talent and experience of these highly skilled teams, it can take several rounds of rework before the developers, testers, and stakeholders agree on the requirements and the scope.

ATDD USING BDD HELPS OVERCOME THE CHALLENGES

ATDD using BDD is a method that the software development triad use to turn requirements into software-development specifications, allowing them to create a non-controversial definition of done. It almost eliminates the chance of misunderstandings. By creating this agreement on the Definition of Done before starting the development and testing efforts, many of the dilemmas and resulting problems listed above are prevented outright.
Behavior-Based Analysis (BBA) is a behavior-based requirements and scope discovery process that the triad uses to precisely identify aspects of the requirement that will or won’t be in scope for the work item the team will be producing. This process enables early identification of critical variations and failure modes that will need to be addressed, which can then be specified using ATDD with BDD to gain upfront agreement on the outcomes desired by the customer/stakeholder. By using this process, many of the dilemmas and resulting problems listed above are also prevented outright.

ATDD with BBD and BBA represent the epitome of Lean and Agile software requirements. It equips the triad with a sustainable approach to

- Build exactly what is being asked, and only that
- Build quality in from the start
- Build the smallest pieces of business value possible
- Increase accuracy and predictability of estimates

It results in high quality acceptance tests that the developers can use to drive their daily activities.

COURSE OBJECTIVES

This workshop combines training and coaching to equip the software development triad.

- Pre-workshop: Half-day meeting with first level management and team-leads to become familiar with your organization and team’s needs.
- Training: 1.5 days. Principles of Agile requirements; ATDD, BDD, and BBA; Done criteria, estimation
- Coaching: 3 hours for each team on a 1-on-1 basis spread over the length of the workshop. Putting principles learned into practice on their backlog.

WORKSHOP OUTLINE

MODULE 1

Training. Everyone together for 4 hours

- Agile requirements game. Learning to see the problems with requirements analysis, management, conveyance, and collaboration. Demonstration of the essential aspects, iterative requirements, fast feedback, and domain specific language.
- Exploration of
  - The fundamentals of Acceptance Test-Driven Development (ATDD) and Behavior-Driven Development (BDD)
  - The importance of collaboration on specifications
  - The importance of having a non-controversial definition of done

- Students get to pick something from their backlog that they are going to work on during the course, and they start doing some rudimentary exercises on it during the morning

Coaching. Each team must get 1 to 1.5 hours of coaching

- The team continues applying the ATDD with BDD techniques on their backlog
- Answering questions, guiding, and coaching them, but just on the aspects learned so far
- Introduction of 1) separation of concerns, 2) higher-level scenarios, and 3) chains of justification

MODULE 2

Training. Everyone together for 4 hours.

- Behavior-Based Analysis (BBA) process
- Collaborating on discovering additional scope around the backlog items they started with in Module 1
- Managing the growing body of knowledge around their selected backlog items while maintaining focus
- How to integrate the analysis process and responsibilities into the team's development process, such as Scrum or SAFe-Team

Coaching. Each team must get 1 to 1.5 hours of coaching

- Learning how to use the BBA process, but with 1:1 coaching and guidance
- Practicing separation of concerns and crafting higher-level vs. lower-level scenarios
- Introduction to the concept of chains of justification,
- How to define the scope of the backlog item they started with, using the newly discovered body of knowledge
- Creating a noncontroversial Definition of Done
- How to estimate using this new understanding of the backlog item’s scope

MODULE 3

Training. Everyone together for 4 hours

- Business Architecture and Capability Management and how to use this as a structure and backdrop for their ATDD using BDD specifications
- How to build thin vertical slices using the Capability structure and ATDD using BDD specifications
- The difference between a) the Capability structure and ATDD using BDD specifications, and b) work management artifacts, i.e. backlog items, such as Product Backlog Items, Epics, Features, User Stories
- How to craft Definitions of Done and Ready, integrate it into Scrum, and continuously improve it
Coaching. Each team must get 1 to 1.5 hours of coaching

- Finalizing the structure of their newly documented body of knowledge
- Next steps in adoption of Scrum and ATDD with BDD, especially focusing on things they can start with immediately. Here are examples of what is covered
- Agreeing on a Definition of Done and a team-based estimation for each backlog before starting development
- ATDD using BDD and BBA for a single scenario and see it through fully to Done in a sprint, gradually using it for more and more scenarios, eventually for an entire backlog item, and finally for an entire Sprint’s worth of backlog items
- Deferring all work requests to the Scrum Product Owner or whomever has the “Team Backlog Management Authority”

FOLLOW-UP

This workshop can be followed up with one to two days of coaching to show you how to automate the testing of ATDD specifications.

LEVEL

Foundational

TARGET AUDIENCE

Members of the software development triad: Customer/Business stakeholder, developer, and tester.

Additional students include Lean-Agile coaches, Product Managers, and Product Owners.

ROOM SETUP AND EQUIPMENT

Tables seating up to 4-5 people

Each table should have whiteboard and flip chart plus a flip chart and whiteboard for the instructor

Projector with screen

COURSE MATERIALS

Course materials will be provided at the start of the class. Attendees should bring existing stories from their current projects to work on as exercises.

PREREQUISITES

None

Consider taking the online course Agile Product Management on the Net Objectives University: https://university.netobjectives.com/courses/agile-product-management-essentials

COURSE LENGTH

Variable

MAXIMUM NUMBER IN CLASS

75

NET OBJECTIVES

We are committed to delivering the principles, practices, and perspectives that businesses must know in order to maximize their return on their technology solution and software development efforts. We combine our experience and a time proven approach based on lean thinking to continuously extend the capability of what is possible in creating effective technology delivery organizations (IT or product). We provide these learned methods to our clients to assist them in achieving their goals and in assisting them in making their organizations more successful.

Full course descriptions may be found at www.NetObjectives.com/training