

What Happened to My Day Job?

Quality Assurance in Agile / Scrum / Kanban

Angela Johnson, PMP, CSM, CSPO, CSP
Aequitas Consulting Ltd

Angela Johnson

Senior Trainer and Coach
Aequitas Consulting Ltd



AequitasConsultingLtd@gmail.com

<http://www.scrumalliance.org/profiles/32158-angela-d-johnson>

Presenter Background

- ▶ Application Development / Technology 15 years
- ▶ Project Manager 11 years specializing in Application Development projects; last 4 years primarily Scrum / Agile projects
- ▶ Project Management Professional (PMP) 5 years
- ▶ Certified ScrumMaster (CSM) 3 years
- ▶ Certified Scrum Product Owner (CSPO) 6 months
- ▶ Certified Scrum Professional awarded Feb 2011

©2011 Aequitas Consulting Ltd

3

Traditional Design & Development

- ▶ The product is considered to be one project, one release, all parts having equal value with a phased or gated approach:
 - Specification
 - Design
 - Code
 - Test
 - Release
- ▶ No business value is delivered until everything is done

©2011 Aequitas Consulting Ltd

4

Why do we Test?

- ▶ Validation is Required
- ▶ Validate requirements understanding
- ▶ Validate meeting the requirements
- ▶ Validate product quality

©2011 Aequitas Consulting Ltd

5

What is Agile Exactly?

- ▶ More of a framework or a philosophy
- ▶ It is a way to think about product development
- ▶ It is not “prescriptive” – there is no such as “THE” Agile method
- ▶ To “be Agile” you need to put the values and principles into practice
- ▶ Processes like Scrum and eXtreme Programming support the Agile philosophy

©2011 Aequitas Consulting Ltd

6

What is Agile Exactly?

- ▶ Scrum: Ken Schwaber, Jeff Sutherland
- ▶ eXtreme Programming (XP): Kent Beck, Ward Cunningham, Ron Jeffries
- ▶ Crystal: Alistair Cockburn
- ▶ Lean Software Development: Mary & Tom Poppendieck
- ▶ Feature Driven Development (FDD): Jeff DeLuca
- ▶ Kanban: Taichi Ohno (Toyota)
- ▶ Others?

©2011 Aequitas Consulting Ltd

7

Agile Manifesto

We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

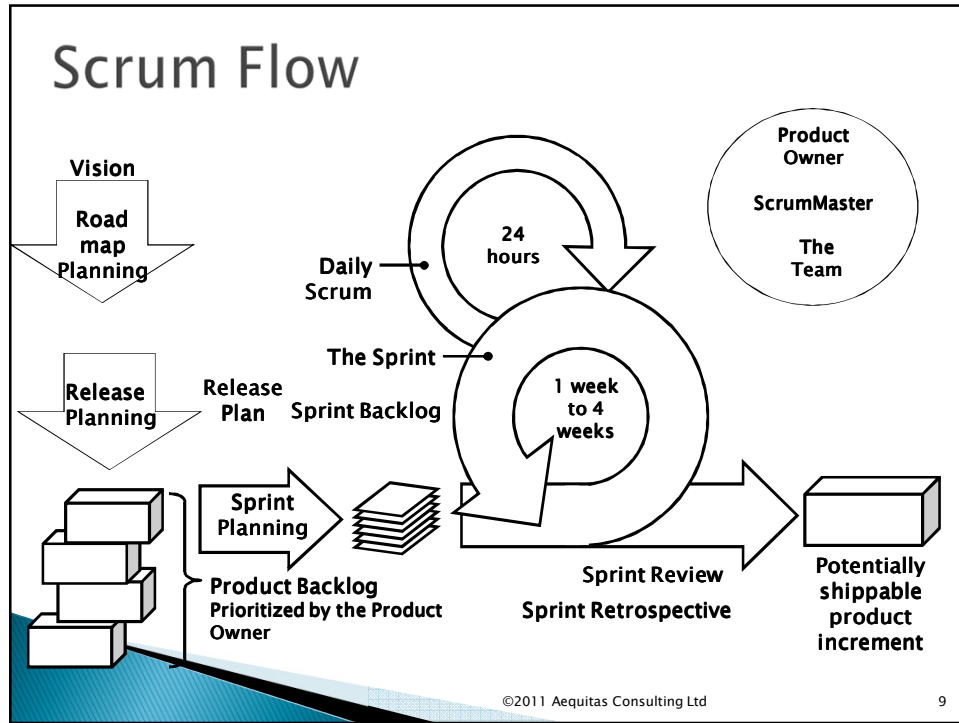
- ▶ Individuals and interactions over processes and tools
- ▶ Working software over comprehensive documentation
- ▶ Customer collaboration over contract negotiation
- ▶ Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left more.

- ▶ <http://agilemanifesto.org/>

©2011 Aequitas Consulting Ltd

8



Remember Validation?

- ▶ Validation is Required
 - Validate requirements understanding
 - Validate meeting the requirements
 - Validate product quality

- ▶ This means must push testing up in the process
 - Improve the conversation between stakeholders and the development team
 - Tests become executable specifications

Pushing Testing Up in the Process

Testing Could Be...

- Attempting to find defects (might be WASTE)
- Attempting to prevent defects (is REQUIRED)

As QA/Testers

- ▶ The goal should *not* be to find defects
- ▶ The goal should be to prevent defects
- ▶ A quality process builds quality into the code

©2011 Aequitas Consulting Ltd

11

Preventing Defects

- ▶ The job of tests, and the people that develop and run tests, is to *prevent* defects, not to find them.
- ▶ A quality assurance organization should champion processes which build quality into code from the start, rather than test quality in later.

Implementing Lean Software Development: From Concept to Cash,
by Mary and Tom Poppendieck. Addison-Wesley 2006

©2011 Aequitas Consulting Ltd

12

Preventing Defects

- ▶ This is not to say that verification is unnecessary. Final verification is a good idea; it's just that finding defects should be the exception, not the rule, during verification.
- ▶ If verification routinely triggers test-and-fix cycles, then *the development process itself* is defective.

Implementing Lean Software Development: From Concept to Cash,
by Mary and Tom Poppendieck. Addison-Wesley 2006

©2011 Aequitas Consulting Ltd

13

So Now What?

- ▶ It is a paradigm shift to make in moving from gates phases to sprints or iterations
- ▶ This can be a bit of a culture shock in going from waiting up to 6 months to incrementally testing within every couple of weeks
- ▶ There is value in testing early, often and involving the whole team
- ▶ Programmers may make use of Test Driven Development writing tests then a bit of functionality, sees it fail, making it pass, etc.

©2011 Aequitas Consulting Ltd

14

What Else does this Mean?

- ▶ The team will have to agree on a “definition of done” with development being done to a Product Owner’s satisfaction
- ▶ Organizational changes may be necessary to support continuous integration and releases to avoid mini-waterfall or “Scrumfall”
- ▶ Quality needs to be moved up in the process to avoid the accumulation of Technical Debt

©2011 Aequitas Consulting Ltd

15

History of FIT – Acceptance Testing

- ▶ First there was FIT, the Framework for Integrated Testing from Ward Cunningham
- ▶ Then came FitNesse a wiki based test management system that uses FIT to run the tests
- ▶ Then came a lot of other FIT implementations, some better than the original...but they were all different

©2011 Aequitas Consulting Ltd

16

FitNesse

- ▶ FitNesse was intended to be a business tool, but it was impractical for developers to use because it was hard to debug into the tests and the system under test
- ▶ FitNesse hands off the processing of the test definition in tables to the FIT server
- ▶ The architecture of FIT is poor; it uses an inheritance for specialization model

©2011 Aequitas Consulting Ltd

17

Simple List Invocation Method

- ▶ Slim is an alternative to FIT
- ▶ Slim keeps the HTML processing, comparisons and results output in FitNesse
- ▶ Slim is easy to port and because all the work is done in FitNesse, the ports will be as nearly identical as the platform allows

©2011 Aequitas Consulting Ltd

18

An Example of Slim

Should I Buy Milk?

If I don't have any cash and I forgot to pay the credit card bill, even if there is no milk left, I can't go to the store for more. If I have at least \$10 and there is no milk, I can go to the store for more. If I don't have any cash but the credit card is good to go, and I am out of milk, I can go get more. If I have at least \$10 and the credit card is good and I am out of milk, I can go get more.

Typical example: `Slim_example.ShouldBuyMilk`

Should I Buy Milk			
cash in wallet	credit card	pints of milk remaining	go to store?
0	no	0	no
10	no	0	yes
0	yes	0	yes
10	yes	0	yes
0	no	1	no

When the Test Runs

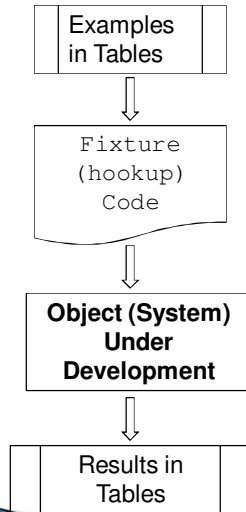
Should I Buy Milk?

If I don't have any cash and I forgot to pay the credit card bill, even if there is no milk left, I can't go to the store for more. If I have at least \$10 and there is no milk, I can go to the store for more. If I don't have any cash but the credit card is good to go, and I am out of milk, I can go get more. If I have at least \$10 and the credit card is good and I am out of milk, I can go get more.

Typical example: `Slim_example.ShouldBuyMilk`

Should I Buy Milk			
cash in wallet	credit card	pints of milk remaining	go to store?
0	no	0	no
10	no	0	yes
0	yes	0	yes
10	yes	0	yes
0	no	1	no

Requirements by Example



- ▶ Tables record business process conversation decisions
 - Names
 - Rules
- ▶ Fixture code connects the examples to the implementation
- ▶ Reports measure progress

©2011 Aequitas Consulting Ltd

21

Test Results

- ▶ Identical to the Slim tables except
 - Output cells
 - Green if fixture returns expected result (pass)
 - Red if fixture returns any other result (fail)
 - Expected value
 - Actual value
 - Yellow if the fixture cannot process the inputs or outputs (error)

Green

Red

Yellow

©2011 Aequitas Consulting Ltd

22

Timeline for Acceptance Tests

- ▶ Product Owner refines stories and acceptance tests from Release Planning meeting, a few days before the Iteration Planning meeting
- ▶ Developers/Testers collaborate adding more detail tests in the Iteration Planning meeting
- ▶ Developers/Testers continue to fill in detail during the Iteration – failing tests until code is implemented

©2011 Aequitas Consulting Ltd

23

Timeline for Acceptance Tests

- ▶ Developers get tests to pass
- ▶ Becomes part of the Regression test suite when story is accepted
- ▶ For more info on Acceptance Testing with FitNesse, visit:
<http://fitnesse.org/FitNesse.UserGuide.AcceptanceTests>

©2011 Aequitas Consulting Ltd

24

Beware of Dogma

- ▶ Dogma is the unshaken belief that something is true, regardless of proof (www.wisegeek.com)
- ▶ It is the authoritative and not to be disputed, doubted or diverged from by the practitioners or believers (Wikipedia)

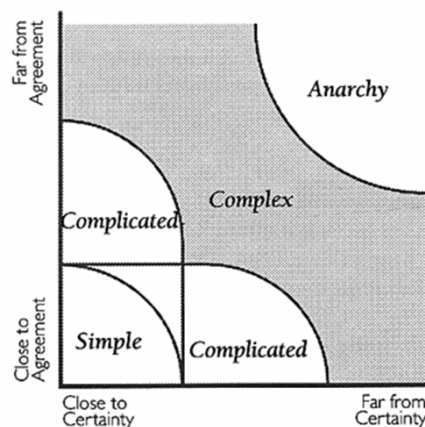


©2011 Aequitas Consulting Ltd

25

The Stacey Diagram

- ▶ Simple projects may not need the overhead of Agile
- ▶ Product Development work tends to fall into the Complex Space
- ▶ The Anarchy Space requires Agile Analysis and Design Pattern understanding to move into the Complex Space
- ▶ Agile returns the biggest process gains in the Complex Space



©2011 Aequitas Consulting Ltd

26

Resistance to Change

- ▶ Resistance is very common
- ▶ It is much easier to fall back on the familiar than to risk something new...especially when you are crunched for time
- ▶ The fear of failing or the “blame game” can also keep you in the status quo

©2011 Aequitas Consulting Ltd

27

Common Failures in Scrum

- ▶ Lack of an empowered, responsive Product Owner is the most common cause of failure
- ▶ Scope creep within a Sprint
- ▶ Not empowering the team to develop its own process to meet their realities...
- ▶ Lack of Automated Testing
- ▶ Ignoring the “People” factor

©2011 Aequitas Consulting Ltd

28

The Quality Role Remains Largely the Same...

- ▶ Focus is still on delivering a quality product
- ▶ Requirements are clarified
- ▶ Tests are created for both functional and non-functional activities
- ▶ Process flaws are identified
- ▶ The business needs and requests are represented

©2011 Aequitas Consulting Ltd

29

Yet with Noted Differences...

- ▶ Work is now completed faster within Sprints, Iterations or Increments
- ▶ The way we communicate has changed completely
- ▶ QA must JOIN the conversation
- ▶ The way QA is approached and approaches the project requires a Paradigm shift
- ▶ The Focus is on the prevention of defects

©2011 Aequitas Consulting Ltd

30

How can I be a Successful Member of an Agile Team?

“Skills are important but attitude counts more.”

Agile Testing: A Practical Guide for Testers and Agile Teams
by Lisa Crispin and Janet Gregory

©2011 Aequitas Consulting Ltd

31

10 Principles for Agile Testers

1. Provide continuous feedback
2. Delivery value to the customer
3. Enable face to face communication
4. Have courage
5. Keep it simple

Agile Testing: A Practical Guide for Testers and Agile Teams
by Lisa Crispin and Janet Gregory

©2011 Aequitas Consulting Ltd

32

10 Principles for Agile Testers...Continued

6. Practice continuous improvement
7. Respond to change
8. Self organize
9. Focus on people
10. Enjoy

**Agile Testing: A Practical Guide for Testers and Agile Teams
by Lisa Crispin and Janet Gregory**

©2011 Aequitas Consulting Ltd

33

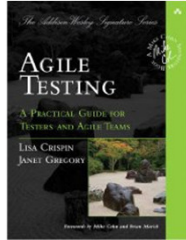
Q & A, Discussion, etc.



©2011 Aequitas Consulting Ltd

34

Agile Testing



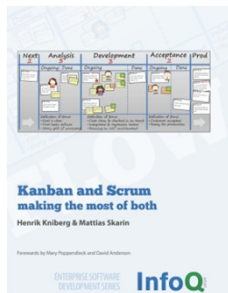
Agile Testing: A Practical Guide for Testers and Agile Teams

by Lisa Crispin and Janet Gregory

©2011 Aequitas Consulting Ltd

35

Kanban and Scrum



Kanban and Scrum: Making the most of both

by Henrik Kniberg & Skarin

©2011 Aequitas Consulting Ltd

36

Lean Software Development



Lean Software Development

and

Implementing Lean Software Development: From Concept to Cash

Mary and Tom Poppendieck



©2011 Aequitas Consulting Ltd

37

User Stories Applied



User Stories Applied: For Agile Software Development

Mike Cohn

©2011 Aequitas Consulting Ltd

38

Additional Reading

- ▶ Agile Retrospectives, Esther Derby & Diana Larsen
- ▶ Agile Testing: A Practical Guide for Testers and Agile Teams, Lisa Crispin & Janet Gregory
- ▶ Agile Estimating and Planning, Mike Cohn
- ▶ Bridging the Communication Gap: Specification by Example and Agile Acceptance Testing, Gojko Adzic

©2011 Aequitas Consulting Ltd

39

Additional Resources

- ▶ www.ScrumAlliance.org
- ▶ www.AgileAlliance.org
- ▶ www.moutaingoatsoftware.com
- ▶ www.AdvancedTopicsInScrum.com
- ▶ www.xp123.com

©2011 Aequitas Consulting Ltd

40