Quality Assurance in an Agile Environment

from a Tester's Perspective
Introduction

Steve Amstadt

– Centare Agile Practice
– Certified Professional Scrum Developer
– 12+ years experience in Software QA on traditional waterfall and ‘Agile’ projects
– Currently working on a Scrum team
– Steve.Amstadt@centare.com
Centare Group?

- Consulting Services Organization
  - Primary focus on Agile, Mobile and Cloud
- Individual and team based development
- Microsoft Partner in ALM
- Training, mentoring, team building
  - Scrum.org training
  - Best practice training
  - Technical and business leadership strategies
Agenda

• What is Agile
• What is Scrum
• What is Agile testing
• The role of QA in Agile
• Challenges to testing in Agile
• Tools and automation in Agile testing
WHAT IS AGILE?
Agile

• A group of software development methodologies, frameworks and best practices based on:
  – Iterative and incremental development
  – Requirements and solutions evolve through collaboration between self-organizing, cross-functional teams
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.
What is Scrum?

- Scrum, which is grounded in empirical process control theory, employs an iterative, incremental approach to optimize predictability and control risk. Three pillars uphold every implementation of empirical process control.
  - Transparency
  - Inspection
  - Adaptation
What is Scrum?

- Scrum is not a methodology that will make you develop better products.
- Not a “silver bullet.”
- Scrum is simply a framework.
- Tool used to build quality software.
- Tool to increase return on investment.
Simple Scrum

- Daily Standup
- Sprint
- Shippable Increment
- Product Backlog
- Grooming
- Acceptance Testing

*Image Credit: Sam Guckenheimer, Microsoft*
Scrum Roles

- ScrumMaster
- Product Owner
- QA Tester
- UI Designer
- Process
- Product
- Team
- Software Engineer
Scrum Time Boxes

- Release Planning
- Sprint Planning
- Daily Scrum
- Daily Scrum
- Daily Scrum
- Sprint Review
- Sprint Retrospective
- Sprint Planning
- Daily Scrum
- Daily Scrum
- Sprint Review
- Sprint Retrospective
- V1.0
- Sprint Planning
- Daily Scrum
- Daily Scrum
- Sprint Review
- Sprint Retrospective
My First Sprint

- What did I get myself into?
- Overwhelmed and unsure of my role
- How do I write test cases when there are no requirements?
- Testing got pushed to the very end
- Felt intimidated
- Didn’t feel like I could properly contribute to planning
- Wasn’t sure what to report in daily standup
- Feeling of unease
- Outside of my comfort zone
WHAT IS AGILE TESTING?
What is Agile Testing?

• Emphasis on individuals and interactions, working software, customer collaboration
• Responsive to change
• Focus on business value vs. conformance to requirements
• Team Deliverable
# Agile Quality is a Team Deliverable

<table>
<thead>
<tr>
<th>Agile Practice</th>
<th>Benefits</th>
</tr>
</thead>
<tbody>
<tr>
<td>Whole Team</td>
<td>• Quality is everyone’s responsibility</td>
</tr>
<tr>
<td></td>
<td>• Quality is not just testing</td>
</tr>
<tr>
<td></td>
<td>• Testing is a first class citizen</td>
</tr>
<tr>
<td>Continuous Integration</td>
<td>• Developers cannot check in code with failing tests</td>
</tr>
<tr>
<td>Continuous Testing</td>
<td>• Test early and often</td>
</tr>
<tr>
<td></td>
<td>• Bugs found closer to when they are introduced making them easier to fix</td>
</tr>
<tr>
<td></td>
<td>• Testing as features are developed</td>
</tr>
</tbody>
</table>
What is an Agile Tester?

• Embraces change and is a change agent
• Collaborates well with technical and business people
• Uses tests to document requirements and drive development
• Experienced exploratory tester
The 10 Principles for an Agile Tester

1. Provide continuous feedback
2. Deliver value to the customer
3. Enable face-to-face communication
4. Have courage
5. Keep it simple
6. Practice continuous improvement
7. Respond to change
8. Self organize
9. Focus on people
10. Enjoy

Source: Agile Testing- A Practical Guide for Testers and Agile Teams Lisa Crispin, Janet Gregory
## Agile vs. Waterfall

<table>
<thead>
<tr>
<th>Agile</th>
<th>Waterfall</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requirements refined as you go and tests document requirements</td>
<td>Requirements are often bad and poorly written</td>
</tr>
<tr>
<td>Change is embraced</td>
<td>Change requires a change request</td>
</tr>
<tr>
<td>Testing is a first class citizen</td>
<td>Testing is often an afterthought</td>
</tr>
<tr>
<td>Whole team approach to testing</td>
<td>Testing is done by testers</td>
</tr>
<tr>
<td>Lightweight metrics</td>
<td>Metrics are often heavyweight and meaningless</td>
</tr>
<tr>
<td>Cooperative relationship between testers and developers</td>
<td>Testers and developers often have an adversarial role</td>
</tr>
</tbody>
</table>
## Agile vs. Waterfall Continued

<table>
<thead>
<tr>
<th>Agile</th>
<th>Waterfall</th>
</tr>
</thead>
<tbody>
<tr>
<td>Zero tolerance for defects</td>
<td>Defects thrown over the wall</td>
</tr>
<tr>
<td>Defect management tool is optional</td>
<td>Defect management tool is needed</td>
</tr>
<tr>
<td>Trust</td>
<td>Distrust and suspicion</td>
</tr>
<tr>
<td>Continuous testing</td>
<td>Testing compressed</td>
</tr>
<tr>
<td>The team estimates time and effort</td>
<td>Plans are dictated by management</td>
</tr>
</tbody>
</table>
CHALLENGES
Agile Testing Challenges

• Fear
• Old habits
• Us vs. Them mentality
• Unclear role of testers
• Mini-waterfall
• Operational Silos
• Lack of skills and domain expertise
• Not understanding Agile concepts
• Existing processes in the organization
TEST AUTOMATION IN AGILE
AUTOMATION IS ESSENTIAL
Why Automate

• Provides safety net
• Gives feedback early and often
• Documents how the application works
• Potential cost reduction
• Frees people to do their best work
What to Automate

• Repetitive tasks
• Unit tests
• Functional tests
• Load tests
• Data creation
• Stable code
What not to Automate

• Usability testing
• Exploratory testing
• One-Off Tests
• Code that is constantly changing
Visual Studio 2010 Ultimate

• Microsoft Test Manager or Microsoft Visual Studio 2010 can be used to run:
  – Unit tests
  – Coded UI tests
  – Database unit tests
  – Load tests
  – Generic tests

• Tests can be run automatically as part of a build process
Microsoft Test Manager

- Define test effort
- Create and run manual steps
- Record manual steps for playback
  - Useful for automating repetitive tasks performed during testing
  - Action recordings can be used to turn manual tests into automated CodedUI tests
- Testing reports
- Diagnostic data collection
- Exploratory testing
- Defect management
Creating Action Recordings

1. Click the down-arrow on the center group switcher and select Testing Center.
2. In the center group menu bar, click Test.
3. Select the test cases that you want to run.
4. On the toolbar, click Run.
Action Recordings Continued

1. Select Create action recording
2. Click Start Test
3. Mark the result of each step.
4. When you have finished testing, click End Test.

Note the Currently Recording indicator which appears on the application currently being recorded.
Resources

- Software Testing with Visual Studio 2010 - Jeff Levinson
Want more?

- Professional Scrum Product Owner
- Professional Scrum Master
- Professional Scrum Foundations
- Professional Scrum Developer
Q&A
Thank You!