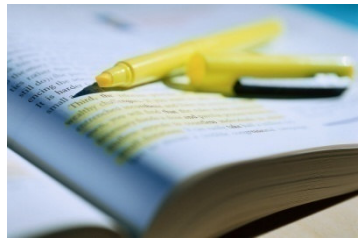
A Case Study in Agile QA

Karen Burley

Engineering Section Manager, HP

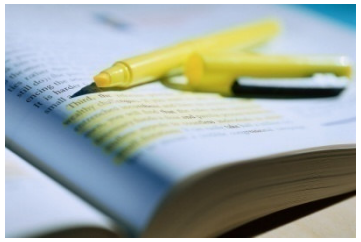
September 13, 2011

In the Beginning…



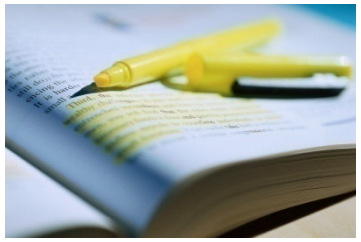
- What is Agile?

In the Beginning…



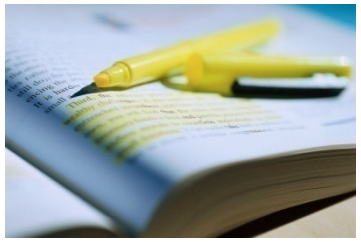
- What is Agile?
- Agile Manifesto: We value
  - **Individuals and interactions** over processes & tools
  - **Working software** over comprehensive documentation
  - **Customer collaboration** over contract negotiation
  - **Responding to change** over following a plan

In the Beginning…



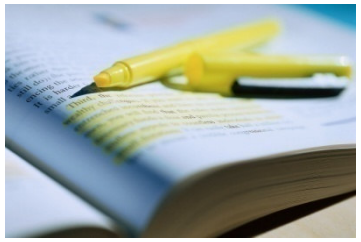
- What is Agile?
  - A toolkit of techniques from which you pick and choose?
  - Examples from Scrum
    - Sprints
    - Scrum Master
    - Daily standup meetings
    - Sprint demos
    - Responding to change

In the Beginning…



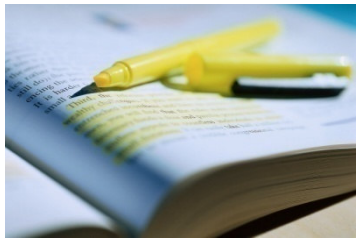
- What is Agile?
  - A toolkit of techniques from which you pick and choose?
  - Where does QUALITY fit?

In the Beginning…



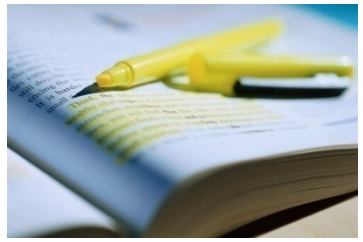
- What is Agile?
  - A toolkit of techniques from which you pick and choose?
  - Where does QUALITY fit?
    - If your Development process is Agile but your QA process is not, what do you have?

In the Beginning…



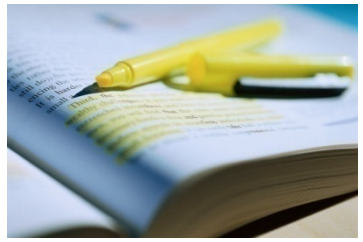
- What is Agile QA?
  - A toolkit of techniques from which you pick and choose?
  - Where does QUALITY fit?
    - If your Development process is Agile but your QA process is not, what do you have?
- In search of quality:
  - Agile Testing, by Lisa Crispin and Janet Gregory

First Attempt…



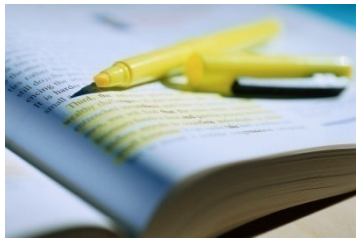
- The role of Agile QA during Sprints

First Attempt…



- The role of Agile QA during Sprints
- What went wrong?

First Attempt…



- The role of Agile QA during Sprints
- What went wrong?
- More Negotiation and the Introduction of a New Process

Shippable Code Process Model

Shippable Code Process Model

- Addressing Product Quality requirements through Process – Based on Agile SDLC Model
- “Shippable Code” Goal
  - **Every day** of the SDLC have working and tested code that can be given to internal or external **beta** customers for evaluation
- Sprint Goal: Delivering working features, at alpha quality, to internal “customers” as soon as possible
- QA & Development partner closely during the Sprint
- Provides early Management visibility into progress

Planning Phase

- Agile SDLC process
  - Goal is to create a Sprint Plan, which includes QA team engagement throughout sprints
- QA:
  - Reviews Requirements Spec and customer use cases
  - Has input into feature scoping & initial design wrt testability
  - Has input into and agrees to project schedule
  - Ensures sufficient resources and hardware for test environment s
  - Provides requirements for test framework, test hooks, API access, error messages, test tools, etc.
  - Has input into customer user interfaces; ensures “If the customer sees it, it has to work right.”
  - Creates project Master Test Plan

Sprint Phase

- During Agile iterations, QA:
  - Participates in functional/design specification reviews
  - Develops feature test plans & system test cases (from use cases) with clearly specified expected test outcomes
    - P0 test = tests basic feature functionality (what beta users would exercise)
    - P1 tests = tests for faults (negative tests), corner cases, stress testing
  - Holds test plan review with stakeholders
  - Optionally, automates test cases
  - Executes P0 system test cases after build passes unit tests
    - Feature marked “DONE” during an iteration if tested successfully
    - Feature that fails testing can not be available to beta users and must be worked on in the next iteration
  - Re-tests all fixes that come in during the Sprint

Sprint Phase

- At end of Sprint:
  - Provides feature test reports
  - Participates in Sprint reviews & gives demos of new functionality
  - Participates in planning for next Sprint
  - Provides traceability of sprint feature to test results
  - Publish metrics: Passed P0 Test Cases
  - Can hand off final Sprint build to internal beta user (who is a proxy for the customer)

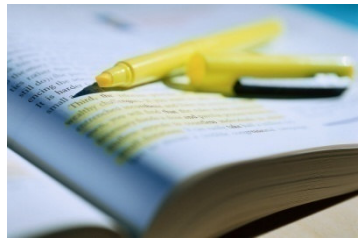
Verification/System Test Phase

- Entrance criteria: Functionality at Beta Quality level
- System Testing occurs during Verification Phase:
  - Execute all P1 tests
  - Integration (of Product components)
  - Integration (Solution testing of multiple products)
  - Regression testing of legacy functionality
  - Stability and stress testing over time
  - Scalability
  - Performance
- Exit criteria: 100% of Test Plan Executed; TBD% of Tests Passed

What Shippable Code Is Not

- Test Automation
- Schedule Accelerator – but it does move up the quality assessment and customer-proxy feedback to an earlier point on the timeline
- A way to produce higher quality code from the start –- but can lead to higher code quality because many defects are found sooner

Recommendation



- Shippable Code Model provides value
  - Early feedback on quality issues
  - Earlier QA involvement means deeper testing, more comprehensive testing
  - Adds most value if product requires a Customer Beta Test
- Shippable Code Model is not cheap
  - QA involved during most of Development Phase *and* all of System Test/Verification Phase
  - If Development needs to rework ➔ QA will need to rework ☹
- Fast Development engagement with QA on problems is key

Key Success Factors in Agile Testing
(from Agile Testing book)

1. Use the Whole Team Approach
2. Adopt an Agile Testing Mindset
   - Proactive, Creative, Open to new Ideas, Collaborate, Willing to take on any Task, Passionate
3. Automate Regression Testing as a Team Effort
4. Provide and Obtain Feedback
5. Build a Foundation of Core Practices
   - Continuous Integration, Std. Test Environments, Manage Technical Debt, Work Incrementally, Coding & Testing are Part of One Process
6. Collaborate with Customers
7. Look at the Big Picture (from Customer's View)

Agile Behaviors that Lead to Success (from Experience)

- Prioritize Features in your Sprint Plan
  - Work on Priority #1 feature in Sprint 1
- Deferred features *must* go into the next sprint
- Plan/load sprint realistically
  - Works best if a “full” sprint is followed by a “light” sprint, to accommodate deferrals
- Development immediately works with QA to troubleshoot problems found during sprint testing
- Line up the engagement with your Internal Customer early!