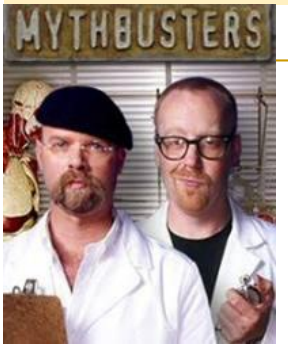
The Keys To Agile Testing Maturity

Myths & Realities from the Trenches



Bob Galen
bob@rgalen.com

Mary Thorn
marythorn@gmail.com

RGalen Consulting Group, LLC
Leading you down the path of Agility...

1

Outline – Myths & Realities

Introduction

1. Transforming your Team
2. **Automation**
3. Developers & Automation
4. Developers Testing
5. **Test Planning & Scripts**
6. **Testing within the Sprint**
7. Exploratory Testing
8. Role of Testers
9. Developer to Tester Workflow
10. **Managing Agile Testers**
11. **Test Metrics**
12. Retrospectives – The Secret Sauce
13. Continuous Improvement
14. The Customer
15. **Agile Requirements – The Product Backlog**

- 3-Pillars of Agile Testing & Quality

RGalen Consulting Group, LLC
Leading you down the path of Agility...

2

#1, Transforming your team

- Myth: You need all programmers or highly technical testers when you move to agile
- Reality: A mix is best –
  - Manual, domain-centric and technical skills
  - Some programming / scripting skills
  - Soft / collaborative skills
- Reality: And throw out all of that Developer-to-Tester ratio 'stuff'.

RGalen Consulting Group, LLC
Leading you down the path of Agility...

3

#2, Automation

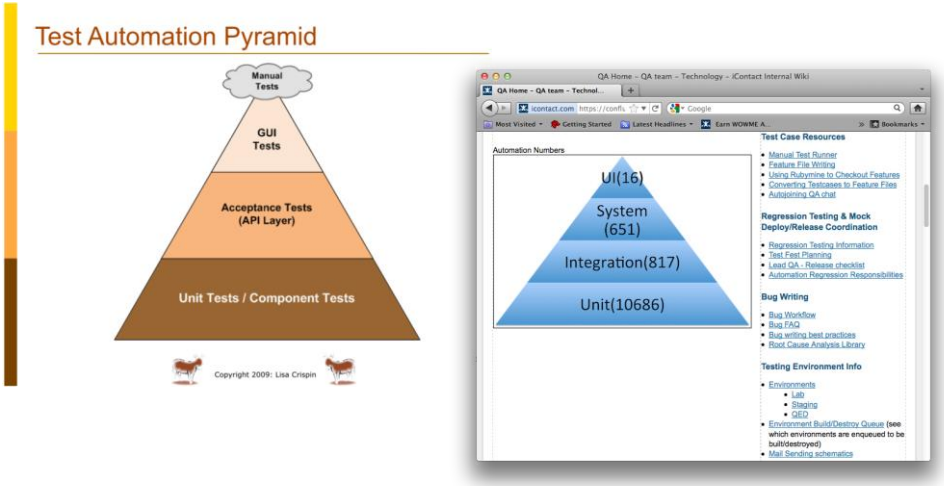
- Myth: You need 100% automation to start agile testing.
- Reality: You simply need to have a strategy AND doggedly pursue automation where it makes sense
  - Make it part of the Backlog and work it every sprint
- Reality: There are some excellent Open Source tools that supplement agile automation development
- Reality: The Agile Test Automation Pyramid is the right overall strategy

RGalen Consulting Group, LLC
Leading you down the path of Agility...

4

Agile Test Automation Pyramid

Mike Cohn; Lisa Crispin & Janet Gregory
http://behaviordrivendevelopment.wikispaces.com/Testing



RGalen Consulting Group, LLC
Leading you down the path of Agility...

Brainstorm…
Agile, Multi-tiered Automation

- Get together in small groups of 4-6 to discuss
- Take a few minutes and think about your current automation approaches:
  - Tooling, approaches & strategies, strengths, weaknesses, opportunities, maintenance challenges, future technology, etc.
- What sorts of adjustments would you need to make to take this approach?
- What would be the largest challenges in taking this approach? How might you overcome them?
- Do you “buy” the whole-team view to automation?

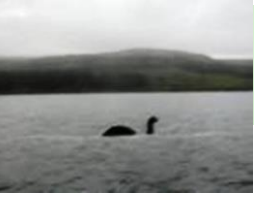
RGalen Consulting Group, LLC
Leading you down the path of Agility...

#3, Developers & Automation

- Myth: QA designs, writes & runs all of the test automation
- Reality: Everyone should be responsible for automation
  - Developers need to minimally attend to Unit Level
  - Participate in any framework or re-use development
  - Writing 'glue' code – fixtures, step files, etc.
- Reality: It also extends into your Build & Continuous Integration systems
  - All automation should be 'wired' into CI
  - Dashboards, trending, lava lamps, etc. for all to see…

7

#4, Developers Testing



- Myth: Developers can't test their own code—they're not independent enough nor skilled enough to do it properly.
- Reality: We need to stop stereotyping team members, their strengths and their abilities.
  - Developers can absolutely test their own code.
  - Some are better at it than others
  - Pair with them to help test appropriately

8

#5, Test Planning & Scripts

- Myth: You don't need to plan
  - (it just happens…)
- and you don't need functional test cases
  - (automation takes care of everything…)

- Reality: Plans help the team focus on the risk-based testing required within an iteration AND across a release
- Reality: Scripts (test cases) help formalize and drive your testing;
  - Absolutely required in regulatory environments
- Reality: You'll never actually automate every test

9

Brainstorm…
Agile Planning & Execution

- Get together in small groups of 4-6

- Take a few minutes discuss your current planning and test process mechanisms.
  - What would an Agile Test Plan "look like" in your organization?
  - What would Test Cases "look like"? What about progress measures? And traceability?
  - Can you move from the "individual" to the "team"?

- Be prepared to share…

10

#6, Testing within the Sprint

- Myth: You simply need to run 100% of the tests within the constraints of the Sprint…that's "Agile"
- Reality: Rarely possible in most contexts.
  - You first need a high-degree of automation and business support (for example: equipment costs)
  - Very mature test automation and CI / CD environments
- Reality: Most agile teams adopt some sort of risk-based testing approach for within the sprints
  - Dealing with Technical Test Debt
  - Then leverage Hardening / Stabilization pre-release sprints

RGalen Consulting Group, LLC
Leading you down the path of Agility...

#7, Exploratory Testing

- Myth: There is no place for Session Based Exploratory Testing in agile contexts.
- Reality: ET and SBET are a beautiful complement to agile testing.
  - Helping nurture pairing & collaboration across teams and functions
  - Defining new (more valuable) test cases
  - Quickly gaining quality & usability feedback
- Let's explore the details of SBET…

RGalen Consulting Group, LLC
Leading you down the path of Agility...

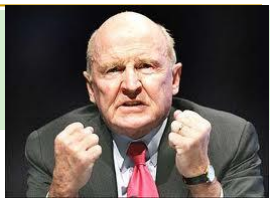
#8, Role of Testers

- Myth: That the testers alone own quality & testing practices within each team and sprint
- Reality: The testers foster a "Whole Team" view towards quality—focusing less on "Testing" and more on "Quality Practices & the Customer"
  - Serving as guides for the team; Testing the "hard bits"
  - Facilitating exploratory testing sessions—finding more interesting / valuable tests
  - Working with the Product Owners—are we solving the customers problems?

#9, Developer to Tester Workflow

- Myth: There is always a hand-off from developers to testers; usually quite late in the sprint. That's simply the "way of things" in software development.
- Reality: Scrummer-fall is alive and well…but, Wrong! Teams need to swarm on their work, as flow & throughput matter the most.
  - WIP limits and close proximity / collaboration help establish a healthy tempo of developer & tester pairing
  - Micro-handoffs – testing as development progresses!
  - Do you log bugs? Or do you fix bugs?

#10, Managing Agile Testers



- Myth: The functional test manager is in charge of deciding how, who, when , etc. for the test team.

- Reality: You still absolutely need functional leadership within agile teams;
  - However, it's focused towards quality practices, strategy & coaching, and handling impediments / escalations
  - Encouraging transparency, transforming metrics & reporting
  - Supporting & protecting the teams
  - Encouraging risk-taking, innovation & creativity (Slack Time)

RGalen Consulting Group, LLC
Leading you down the path of Agility...

15

Levels of Done-Ness Criteria

Activity	Criteria	Example
Basic Team Work Products	Done'ness criteria	Pairing or pair inspections of code prior to check-in; or development, execution and passing of unit tests.
User Story or Theme Level	Acceptance Tests	Development of FitNesse based acceptance tests with the customer AND their successful execution and passing. Developed toward individual stories and/or themes for sets of stories.
Sprint or Iteration Level	Done'ness criteria	Defining a Sprint Goal that clarifies the feature development and all external dependencies associcated with a sprint.
Release Level	Release criteria	Defining a broad set of conditions (artifacts, testing activities or coverage levels, results/metrics, collaboration with other groups, meeting compliance levels, etc.) that IF MET would mean the release could occur.

RGalen Consulting Group, LLC
Leading you down the path of Agility...

16

Brainstorm…
"Your" Definition of Done

- Get together in small groups of 4-6 to discuss
- Using the 4-tier approach referenced start filling in the 4 levels as a group.
- Consider any criteria you are currently using at your companies?
- Also consider current issues or challenge you might have where "done-ness" would help?
- And what about Ready-ness?
- Be prepared to share…

#11, Test Metrics

- Myth: You can and should move forward reporting everything exactly as you have before.
  - Including any 'dysfunctional' metrics that your process and/or PMO dictates.
- Reality: The metrics should change immediately.
  - From QA and Test centric towards Team-Centric metrics (Value, Throughput, Quality, Team)
  - Stop reporting out on "Testing"; it's irrelevant!
  - This effects planning as well—estimation, progress, risk, etc.
  - Contribute quality-centric Information radiators to the mix

Brainstorm…
Morphing your Metrics

- Get together in small groups of 4-6 to discuss
- What are you measuring today? Why?
  - How are they driving your success and behaviors?
- As you move to agile, what can/should you be measuring in the 4 key areas:
  - Value, Quality, Throughput & Predictability, and Team?
- How will you change your existing metrics? What behaviors are you trying to inspire?
- Be prepared to share…

#12, Retrospectives:
The Secret Sauce

- Myth: Testers are “Second Class” citizens who don’t play an active part in the project & team
- Reality: There are many places to “make a difference”
  - Getting the 800 lb. Gorillas out on the table; Showing courage; telling truth
  - Fostering continuous improvement within the team
  - Setting the example; showing vulnerability—admitting you’re wrong
  - Team listening; active planning; dependencies; pairing
  - Risk-taking; Failure!

#13, Continuous Improvement

- Myth: We're generally 'stuck' in our approaches so just accept them and do the "best you can".

- Reality: Continuous improvement is everyone's responsibility—to engage, suggest, take ownership of current results, explore root causes, etc.
  - Active participation in your teams Retrospectives is a key way to guide quality, testing, and customer-centric improvements.
  - Courage!

RGalen Consulting Group, LLC
Leading you down the path of Agility...

#14, The Customer

- Myth: Business Analysts capture customer requirements and testers test them for completeness.

- Reality: You need to begin to partner with the Customer – Stakeholders – Product Owners to produce software that solves the their problems.
  - Move to the "front" and help define & refine User Stories with your Product Owner
  - Actively participate in Sprint Reviews
  - Show value for automation; placing test investments in the Backlog

RGalen Consulting Group, LLC
Leading you down the path of Agility...

#15, Agile Requirements – The Product Backlog

- Myth: We can't start testing until the requirements are finished or stable; no matter how 'agile' we are.
- Reality: Hogwash! Get over it…
  - Ambiguity and incompleteness need to become your friend and ally.
  - As does working with your Product Owners and Customers to help define the requirements
  - Realizing that the requirements (User Stories) are only complete at the end of each sprint.

Brainstorm… Agile Requirements

- Get together in small groups of 4-6 to discuss
- Are iterative, are intentionally incomplete
- The "older" the are, the larger and less defined they are
- Enter the sprint at 70%, exit at 100%
- Drive questions, dialogue, discussion, and collaboration; think 3 Amigos or the Triad
- So, WHY? And how will you make this work as a tester?
- Be prepared to share…

3 Pillars of Agile Quality

Development & Test Automation

- Pyramid-based Strategy: (Unit + Cucumber + Selenium)
- Continuous Integration
- Attack technical infrastructure in the Backlog
- Visual Feedback – Dashboards
- Actively practice ATDD and BDD

Software Testing

- Risk-based testing: Functional & Non-Functional
- Test planning @ Release & Sprint levels
- Exploratory Testing
- Standards – checklists, templates, repositories
- Balance across manual, exploratory & automation

Cross-Functional Team Practices

- Team-based Pairing
- Stop-the-Line Mindset
- Code Reviews & Standards
- Active Done-Ness
- Aggressive Refactoring of Technical Debt
- User Stories, "3 Amigo" based Conversations

- Whole Team Ownership of "Quality"
- Building it 'Right'; Building the 'Right' Thing
- Healthy – Agile Centric Metrics
- Center of Excellence or Community of Practice
- Strategic balance across 3 Pillars; Assessment, Recalibration, and Continuous Improvement

RGalen Consulting Group, LLC
Leading you down the path of Agility...

Foundation of the 3 Pillars

- Whole Team Ownership of "Quality"
- Building it 'Right'; Building the 'Right' Thing
- Healthy – Agile Centric Metrics
- Center of Excellence or Community of Practice
- Strategic balance across 3 Pillars; Assessment, Recalibration, and Continuous Improvement

- Whole team view includes building it right, everyone tests,
- Focus on features/stories, confirmation, conversation, and getting them staged properly OVER testing
- 4-tier metrics: Quality, Value, Prediction, Team
- Agile strategies need light-handed "steering"; establish a CoE (heavier weight) or a CoP (lightweight)
- Consider finding an assessment framework and then tying it to your strategy measurement, recalibration, and continuous improvement.
- Make the Foundation visible thru Information Radiators and metrics

RGalen Consulting Group, LLC
Leading you down the path of Agility...

3 Pillars of Agile Quality

Development & Test Automation

- Pyramid-based Strategy: (Unit + Cucumber + Selenium)
- Continuous Integration
- Attack technical infrastructure in the Backlog
- Visual Feedback – Dashboards
- Actively practice ATDD and BDD

A central part of agile adoption is focusing on CI, 3-tiered Automation development, and Dashboards to begin incrementally building coverage for faster feedback on changes.

In the interim, Hardening or Stabilization Sprints and having a risk-based Release Train concept help

It's important that Test or QA not 'own' the tooling or all of the automation efforts. The strategy can come from Test, but the tactical automation development is best left to the team.

Mature teams invest in automation as part of Done-ness and continually on their backlogs

RGalen Consulting Group, LLC
Leading you down the path of Agility...

3 Pillars of Agile Quality

Software Testing

- Risk-based testing: Functional & Non-Functional
- Test planning @ Release & Sprint levels
- Exploratory Testing
- Standards – checklists, templates, repositories
- Balance across manual, exploratory & automation

Exploratory Testing (Charter / Session based and paired) can be an incredibly effective way to establish a whole-team, collaborative view towards quality and testing. It also emerges new tests.

Leverage 'plans' as a whole-team collaboration mechanism…and do plan.

Do not measure testing or tester progress; instead, measure throughput, output, sprint outcomes, and done-ness escapes at a team level.

You need a balanced test team; not everyone needs to be able to program. But everyone needs to be skilled testers.

Agile testing is a Risk-Based play in every Sprint and across a release sequence. Don't forget your techniques!

RGalen Consulting Group, LLC
Leading you down the path of Agility...

3 Pillars of Agile Quality

Cross-Functional Team Practices

- Team-based Pairing
- Stop-the-Line Mindset
- Code Reviews & Standards
- Active Done-Ness
- Aggressive Refactoring of Technical Debt
- User Stories – 3 Amigo based Conversations

One of the hardest areas to get 'right' culturally. It needs leadership alignment from Quality/Testing to Product to Development and a consistent voice of whole-team approaches.

This is where LEAN lives, where whole-team collaboration happens, where professionalism and craftsmanship are held dear.

I like the view of testers becoming the VOC, champions of quality, and consistent questioners of what is being build. Are we solving the right problems…as simply as possible. Notions of Minimal Viable Product / Feature help with focus.

And yes Virginia, there ARE standards, templates, and a focus on consistency!

RGalen Consulting Group, LLC
Leading you down the path of Agility...

Key Goal of the 3-Pillars… What does "good" look like?

- Whole Team (ownership, accountability, respect, quality)
- Build the right thing…and build it right
- Definition of Done; Ready-ness
- Tackling Technical Test Debt
- Just Enough, Just in Time
- Continuous Improvement
- Commitment to Agility (even when the going is hard)
- Results – Value – Working Code
- Context-based Testing
- 3-Amigos (Team and Organizational levels)
- Feedback, Feedback, Feedback
- Balance

RGalen Consulting Group, LLC
Leading you down the path of Agility...

3-Pillars of Agile Quality & Testing
Thank you!

Development & Test Automation

- Pyramid-based Strategy: (Unit + Cucumber + Selenium)
- Continuous Integration
- Attack technical infrastructure in the Backlog
- Visual Feedback – Dashboards
- Actively practice ATDD and BDD

Software Testing

- Risk-based testing: Functional & Non-Functional
- Test planning @ Release & Sprint levels
- Exploratory Testing
- Standards – checklists, templates, repositories
- Balance across manual, exploratory & automation

Cross-Functional Team Practices

- Team-based Pairing
- Stop-the-Line Mindset
- Code Reviews & Standards
- Active Done-Ness
- Aggressive Refactoring of Technical Debt
- User Stories, "3 Amigo" based Conversations

Whole Team Ownership of "Quality"

Knowing the Right Thing to Build; And Building it Right

Healthy – Agile Centric Metrics

Steering via: Center of Excellence or Community of Practice

Strategic balance across 3 Pillars; Assessment, Recalibration, and Continuous Improvement

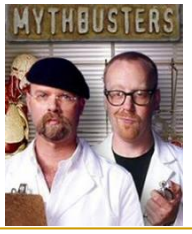
RGalen Consulting Group, LLC
Leading you down the path of Agility...

Wrapping up…

Agile is the best thing that's happened to testers since…
The Great Depression

- Whole Team view
  - Testing, Metrics, Automation
  - Planning, Reporting, Quality
- Facilitate feedback
- Multi-tiered automation
- Just-in-Time, risk-based testing
- Continuous improvement
- Trust the Team

- **Retrospective**



RGalen Consulting Group, LLC
Leading you down the path of Agility...