TECHWELL™

STAR EAST

A TECHWELL EVENT

**W5**
Agile Testing
Wednesday, May 2nd, 2018
11:30 AM

# Thirteen Patterns of Testers Thriving in Agile Teams

**Presented by:**

**Shaun Bradshaw**

**Zenergy Technologies, Inc**

**Brought to you by:**

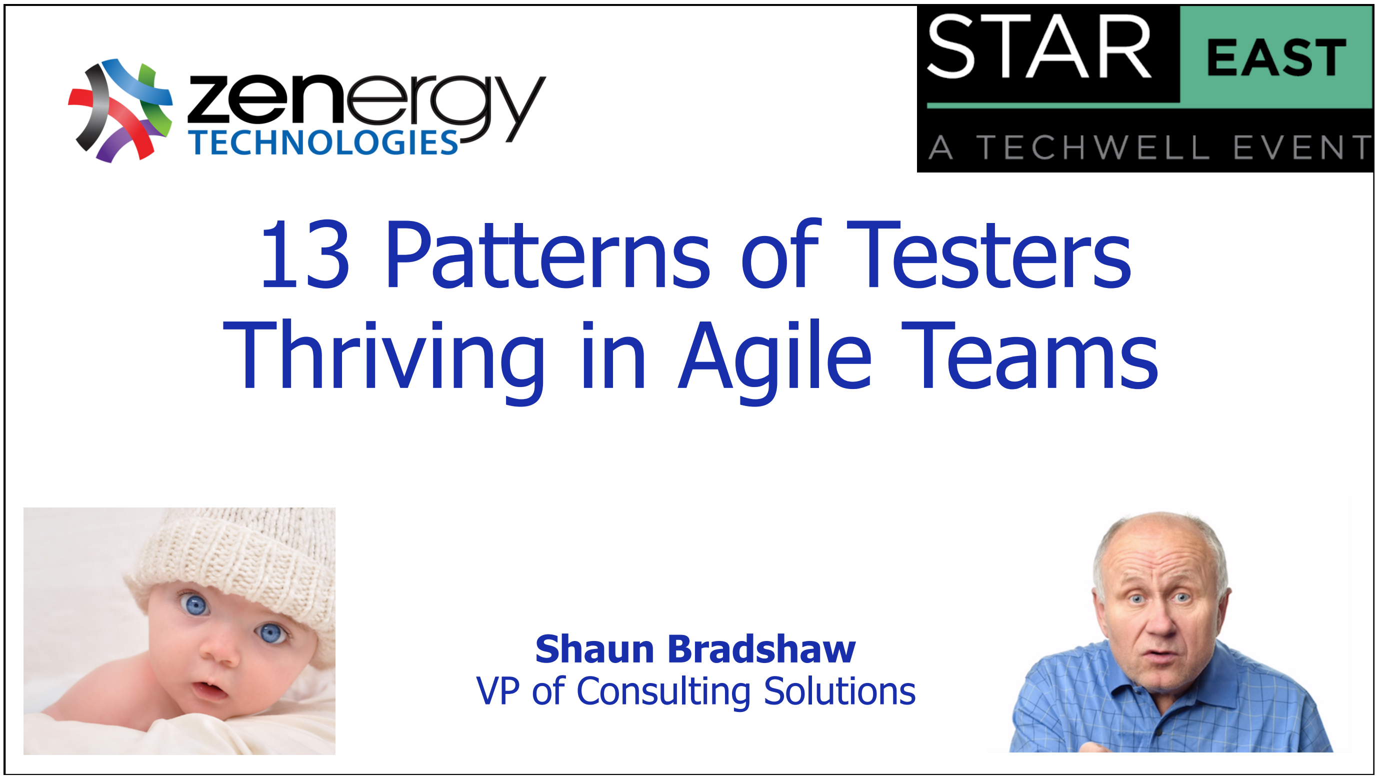
TECHWELL™

350 Corporate Way, Suite 400, Orange Park, FL 32073
888-268-8770 - 904-278-0524 - info@techwell.com - http://www.stareast.techwell.com/

# Shaun Bradshaw

## Zenergy Technologies, Inc

Shaun Bradshaw is a cofounder and principal of Zenergy Technologies, a software delivery solutions firm with multiple industry experts under one roof and a large testing facility in North Carolina. With more than twenty years in the IT industry, Shaun is a recognized expert, coach, and thought leader in QA and test process improvement, agile testing, test management, and metrics. Shaun has architected the test strategy and managed large teams of testers for numerous projects, including a multi-year ERP implementation that went into production with no major issues. More recently, Shaun has spent a majority of his consulting time assisting organizations through their agile transformations, coaching leaders, managers, and teams to ensure they make the shift not just into agile practices, but agile thinking and behaviors.

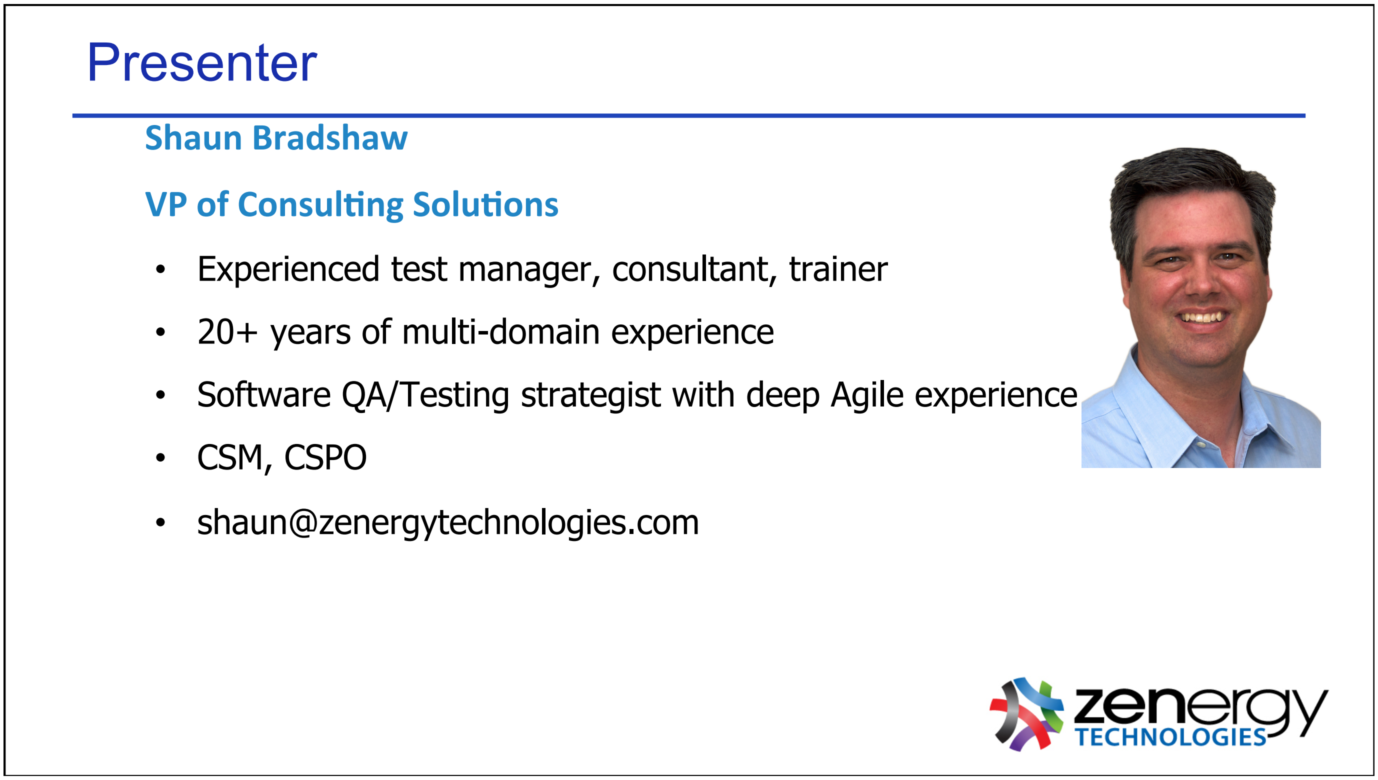


zenergy TECHNOLOGIES

STAR EAST A TECHWELL EVENT

# 13 Patterns of Testers Thriving in Agile Teams

**Shaun Bradshaw**
VP of Consulting Solutions



## Presenter

**Shaun Bradshaw**

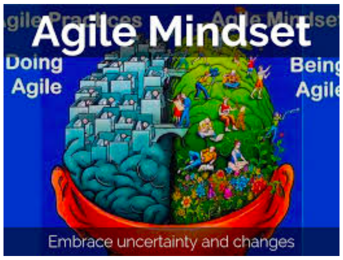
**VP of Consulting Solutions**

- Experienced test manager, consultant, trainer
- 20+ years of multi-domain experience
- Software QA/Testing strategist with deep Agile experience
- CSM, CSPO
- shaun@zenergytechnologies.com

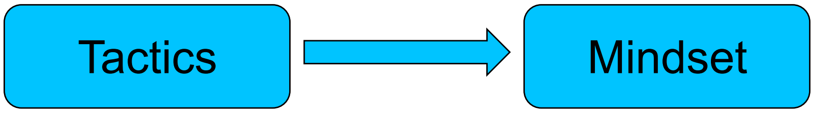
zenergy TECHNOLOGIES

## “Doing” vs. “Being” Agile?

- One debate in the agile community surrounds agile maturity. A way of characterizing it surrounds
  - **Doing Agile** – focusing towards is tactics, ceremonies, and techniques
  - **Being Agile** – focusing towards team mindset, leadership mindset, behaviors, organizational adoption, etc.
- The Mature Patterns workshops crosses both, with emphasis towards the Being-side of the equation.



Embrace uncertainty and changes


Tactics
Mindset

## Agile Testing vs. Traditional Testing

### Traditional

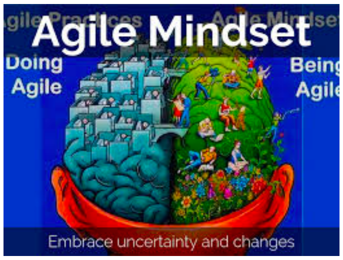
- Testing-focus
- Reliant on detailed requirements and documentation
- Plan-driven approach
- Functionally silo test teams by domain and technology
- Test management tools and Big “A” automation tools

### Agile

- Quality-focus
- Focused on team interaction/ conversations for requirement clarity
- Minimal test plans
- Higher competency across multiple domains and technologies
- Open Source automation models

## The Agile Tester’s Mindset

- Skepticism (versus pessimism)
- Curiosity
- Emotional Intelligence
- Team-oriented
- Learning and Observation
- Persistent
- Try to Break the System



zenergy TECHNOLOGIES

## The Agile Tester’s Perspective

- Must have a combination of:
  - Analytical / Technical skills
  - Customer / Value Perspective
  - Soft / Influence / Communication skills
- Champion of Quality (not the owner)
  - Understand the difference between QA and testing
  - **Communicate** the value of defect prevention and defect detection
  - Expose risk to people who matter, when it matters
  - Rally the team to a QA perspective

zenergy TECHNOLOGIES

## Agile Test Maturity Patterns Outline

1. Ruthless KISS
2. Swarm to the Top
3. Whole Team QA Ownership
4. Quality on ALL Fronts
5. Active Done-Ness
6. Communicate Early and Often
7. Continuously Engage the PO
8. Build Trust with the Developers
9. Test Case Failures – What if its not a bug?
10. Agile Test Automation – aka Flip the Triangle
11. Continuous Learning
12. Yes, There is Planning in Agile
13. Metrics (What to Measure?)

## 1) Ruthless KISS

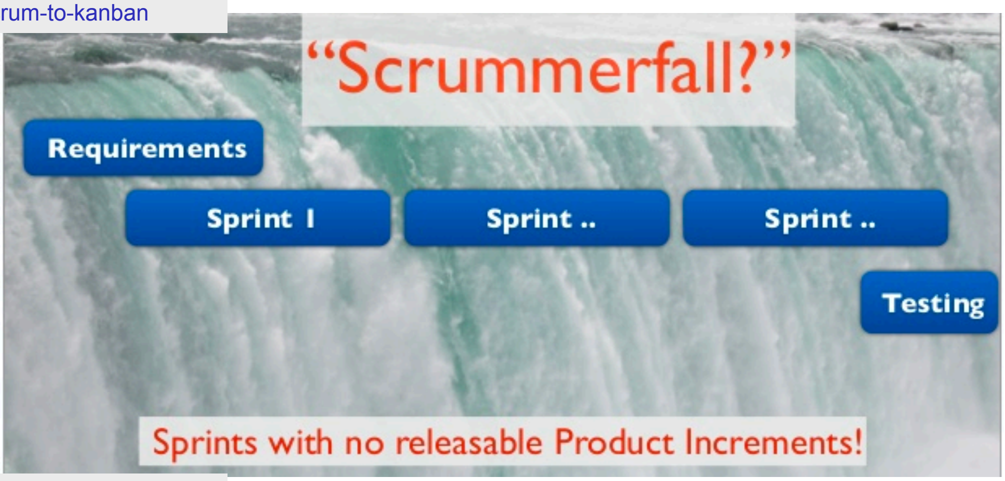
- Get LEAN deep in your DNA
  - Fight Gold-plating your test plans, test cases, and test coverage
- Utilize Acceptance Criteria like a Charter in Exploratory Testing
- Think in terms of MITs – remember there will be other sprints
  - Positive tests first
  - Just enough negative testing
  - Don’t duplicate multi-layered tests (transparency builds trust)

## 2) Swarm to the Top

- Minimize multi-tasking
  - Focus on top stories/tasks
  - Focus on MITs
- Comfortable with on-the-fly test analysis
  - Exploratory Testing
- Document test plans, test cases, and defects only as necessary
  - Test strategy and plans at Release level
  - Tests within the sprint
  - Defects if/when they cross sprints

## Beware Scrummer-fall

By Rachel Davies:
https://www.slideshare.net/RachelDavies/moving-from-scrum-to-kanban



"Scrummerfall?"

Requirements

Sprint 1 | Sprint .. | Sprint ..

Testing

Sprints with no releasable Product Increments!

Our YouTube video:
https://www.youtube.com/watch?v=1LPZa-hbJ2s

# 3) Whole Team QA Ownership

- ➢ Leaving behind the notion that testers "own" quality
- ➢ Create healthy relationships w/
  - Developers (break down the silos)
  - SMs (look to for advice and input)
  - POs (give/receive feedback on AC, test cases, defects)
- ➢ Opportunistic pairing
- ➢ *Don't fear passionate debate & healthy conflict*
- ➢ Stop thinking of "Dev Complete" & "Test Complete"

# 3) Whole Team QA Ownership

- ➢ Create an environment where the whole-team embraces and helps with testing
  - Test Strategies / Designs / Plans
  - All types of test cases (manual, automation, performance)
  - Never letting tests break
  - Pair w/ Dev to build in testability
- ➢ Create a shared QA goal across the team
  - Influence development priorities
  - Negotiate with the PO & Dev team members
- ➢ Ensure test estimates are part of work estimation
- ➢ Perform Root Cause Analysis as a team

# 4) Quality on ALL Fronts

- ➢ Rally the team to focus on defect prevention not just defect detection
- ➢ Cultivate professionalism within the team
  - Doing the right things…doing things right (design inspections, requirements discussions, code reviews, etc.)
  - Shift-Left Thinking
  - Alter team's mindset and actions from I-shaped to T-shaped
- ➢ Encourage self-inspection; self-policing
- ➢ Focus on *Craftsmanship and Professionalism*

zenergy TECHNOLOGIES

# 5) Active Done-Ness

As a tester what does "I'm done with the story" mean?

- ✓ Test cases designed with a broad view to test cases (unit, functional, acceptance, performance, regression)
- ✓ Test cases pair-reviewed with dev & test team members
- ✓ Test cases - checked into repository
- ✓ All test cases tied to Acceptance Criteria have been automated and passed
- ✓ Test automation built into Continuous Integration environment

zenergy TECHNOLOGIES

## 6) Communicate Early and Often

- Identify questions/concerns in stories, estimates, tasks, etc.
- Embrace the 3 Amigos
- Active Pairing w/ Dev
  - What should be tested
  - Who will test
  - How should it be tested
  - What data is necessary
- Blockers and impediments
  - Don't wait for the stand-up
  - Ask for help (PO, SM, Dev, anybody on the team really…)

By Frits Ahlefeldt

## 3 Amigos: Dev + Test + Product

- Are often used as a metaphor for improved backlog refinement
  - 3-Amigo meetings
  - Story Owners or shepherd
- Multi-perspective conversations during the life-cycle of the story
  - From Concept (Epic) to Story delivery - done
- Doesn't always limit to 3 perspectives

## 7) Continuously Engage the PO

- Make the PO your new BFF
- Get to know the "why" behind the stories
- Help develop the acceptance criteria – influence as necessary
- Focus on his/her priorities using that input to inform a risk-based testing approach
- Get his/her input on defects
  - What's the defect priority? Effort? Focus?

- Voice of the customer
- Understanding value proposition

zenergy TECHNOLOGIES

## 8) Build Trust with the Developers

- Ask questions – learn what they do and how they do it

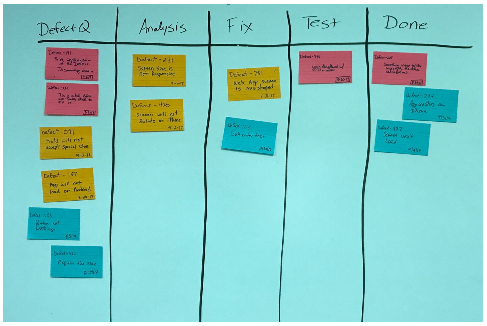
- Ways to build trust
  - Don't be a chicken little
  - Don't cry wolf
  - Don't call their baby ugly
  - Take responsibility
  - Investigate issues

- Communicate, communicate, communicate

Sorry I said your baby was ugly. What I meant was your baby is uniquely unattractive.

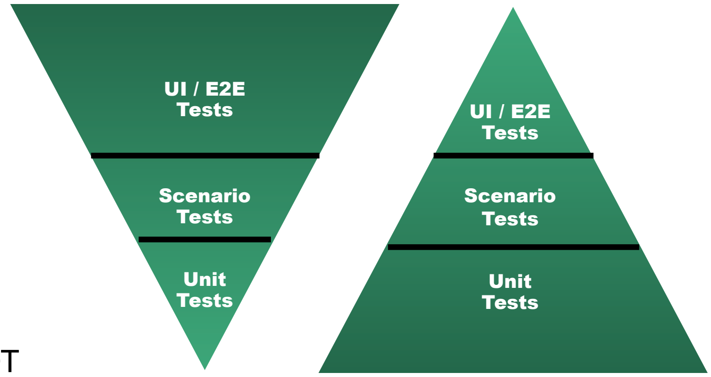
## 9) Test Case Failures – What if its not a bug?

- If a test fails, did you find a defect?
  - Can the failure be duplicated?
  - Was the test properly executed?
  - Was the failure due environmental or data issues/configurations?
  - What error message was generated?
  - What is the nature of the failure and what are the potential causes?
- Assume the failure isn't a bug until you can prove otherwise
- When you find a defect
  - Conversations first and documentation second
  - White board & sticky before tool

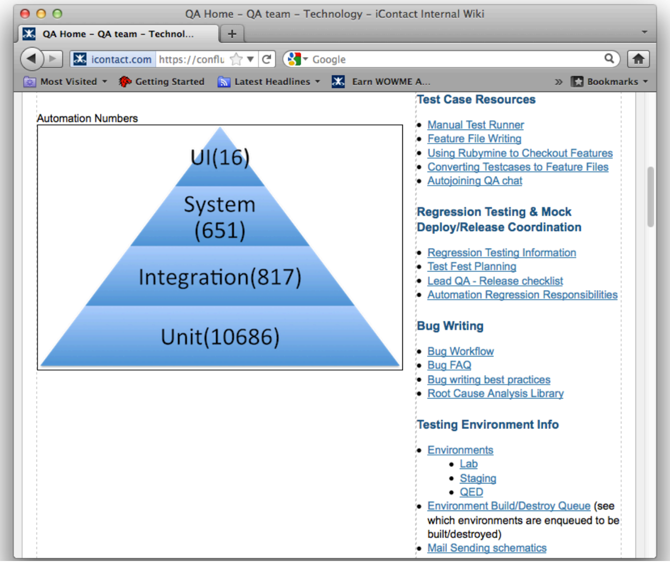


## 10) Agile Test Automation – aka Flip the Triangle

- Invest in test automation (part of DoD)
- Test Automation Focus shifts to
  - Lots of unit tests (TDD)
  - Some scenario-based, API tests (BDD)
  - Few UI (Traditional)
- Key goal is continuous & fast feedback
  - CAUTION: 100% automation is NOT the goal



## Agile Test Automation Pyramid - Mike Cohn; Lisa Crispin & Janet Gregory



Automation Numbers

**Test Case Resources**

- Manual Test Runner
- Feature File Writing
- Using Rubymine to Checkout Features
- Converting Testcases to Feature Files
- Autojoining QA chat

**Regression Testing & Mock Deploy/Release Coordination**

- Regression Testing Information
- Test Fest Planning
- Lead QA - Release checklist
- Automation Regression Responsibilities

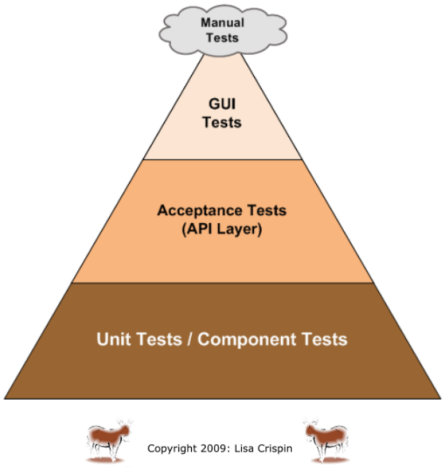
**Bug Writing**

- Bug Workflow
- Bug FAQ
- Bug writing best practices
- Root Cause Analysis Library

**Testing Environment Info**

- Environments
  - Lab
  - Staging
  - QED
- Environment Build/Destroy Queue (see which environments are enqueued to be built/destroyed)
- Mail Sending schematics

Test Automation Pyramid



Copyright 2009: Lisa Crispin

## 11) Continuous Learning: Yours + Team

- 90% of testing remains the same
- Determine what you don't know and create "learning goals"
  - Sprint 1 – how scrum works
  - Sprint 2 – how to estimate all work
  - Sprint 3 – database development
  - Sprint 4 – automation
- Think in terms of Shu Ha Ri
- Identify a mentor and/or establish a Community of Practice around
  - Agile, Test Automation, Testing (plans, designs, cases, etc.)

守破離

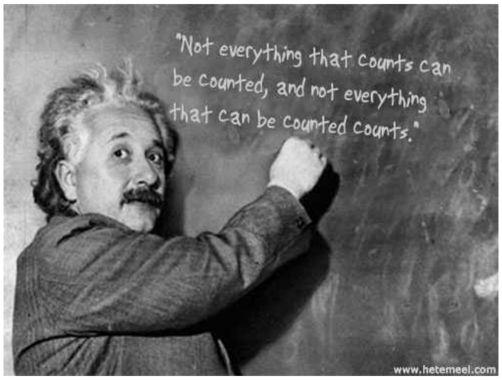
# 12) Yes, There is Planning in Agile

- Apply Risk-Based Testing techniques to all of your team's testing
  - Daily level
  - Sprint level
  - Release level
- Plan test strategy as a team
  - Part of Sprint Planning
  - Release (PI) Planning
  - Who's plan is it?
- The plan is irrelevant; whole, agile team planning is everything.

zenergy TECHNOLOGIES

# 13) Metrics, i.e. What to Measure?

- Traditional metrics measured test team and tester:
  - Test cases, coverage, bugs, time, etc.
- Don't do that any more. Now it's about the TEAM!
- Measure:
  - Velocity, Flow, Throughput, Predictability
  - Escapes, DoD exceptions, story slips
  - Value delivered, ROI, customer satisfaction
  - Team happiness


"Not everything that counts can be counted, and not everything that can be counted counts."
www.hetemeel.com

zenergy TECHNOLOGIES

## Wrap-up

- What were the most compelling patterns?
- What essential patterns did we miss?
- Final questions or discussion?

Thank you!

zenergy TECHNOLOGIES

## Contact Info

**Zenergy Technologies | 336.245.4729 | Zenergytechnologies.com | contact@zenergytechnologies.com**

**Shaun Bradshaw**
shaun@zenergytechnologies.com
**@shaunbradshaw**

zenergy TECHNOLOGIES