

# Agile Testing Practices

University of Calgary, February 2011

Janet Gregory, DragonFire Inc.

Copyright 2010



With material from Lisa Crispin

# Topics

---

- A Little About a Tester's World
- Testing and Agile
- Types of Testing
- Wrap-up



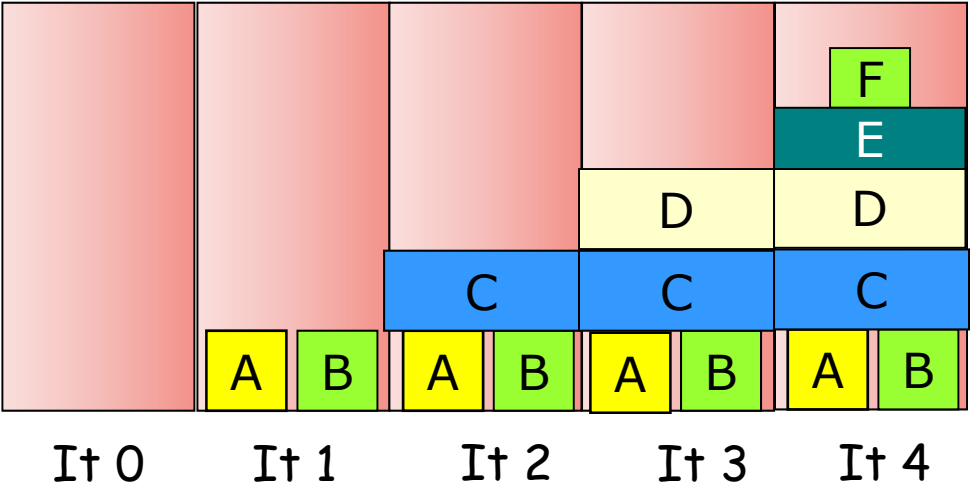
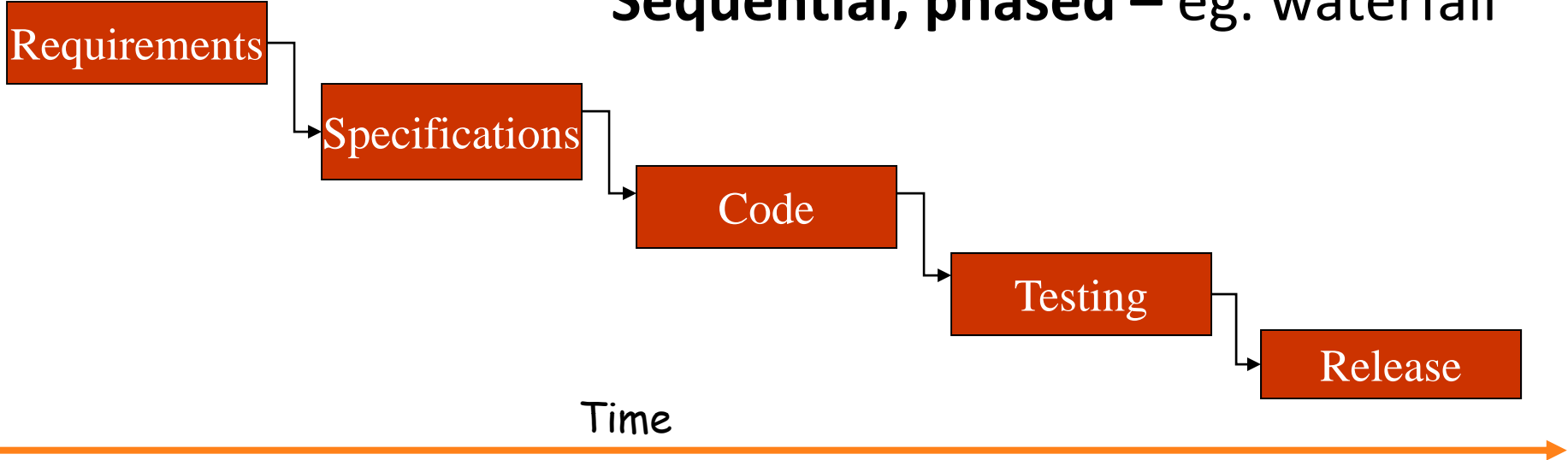
# Agile is a term to describe methodologies that:

- have short iterations
- encourage active customer participation
- demand whole team collaboration
- test features as they are coded
- deliver business value at regular intervals
- adapt their processes based on feedback

...and so on.....



# Sequential, phased – eg. waterfall



## Agile: iterative, incremental

- Each story is expanded, coded and tested
- Possible release after each iteration



# Agile Testing – What is It?

---

- Testing is an activity, not a phase
- Collaborating with customers & developers
- The whole team is committed to quality
- The whole team is test-obsessed
- Focus is on the prevention of bugs,
  - Not detection



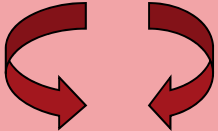
# Who are Agile Testers?

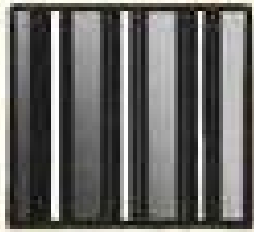
---

- Integral part of the team
- Drive development with tests
- Work with customers to define acceptance tests for each story / feature
- Provide continuous feedback to the team
- Provide constructive skepticism
- Test each story as it is complete



# Test Approach – The Agile Way

Project Initiation	Get an understanding of the project (business)	
Release/Project Planning	Participate in sizing stories	Create Test Plan
<p>Each Iteration</p> <p>1</p> <p>...</p> <p>X</p> 	<p>Estimate tasks, Run regression tests</p> <p>Collaborate with customers on acceptance tests</p> <p>Write, automate and execute new story tests</p> <p>Pair test with other testers, developers</p> <p>Perform exploratory testing</p>	
System Test / End Game	<p>Perform Load Test</p> <p>Complete Regression Test</p> <p>Perform UAT</p> <p>Perform Mock Deploy</p> <p>Participate in Release Readiness</p>	
Release to Prod/ Support	<p>Participate in Release to Prod</p> <p>Participate in Retrospectives</p>	

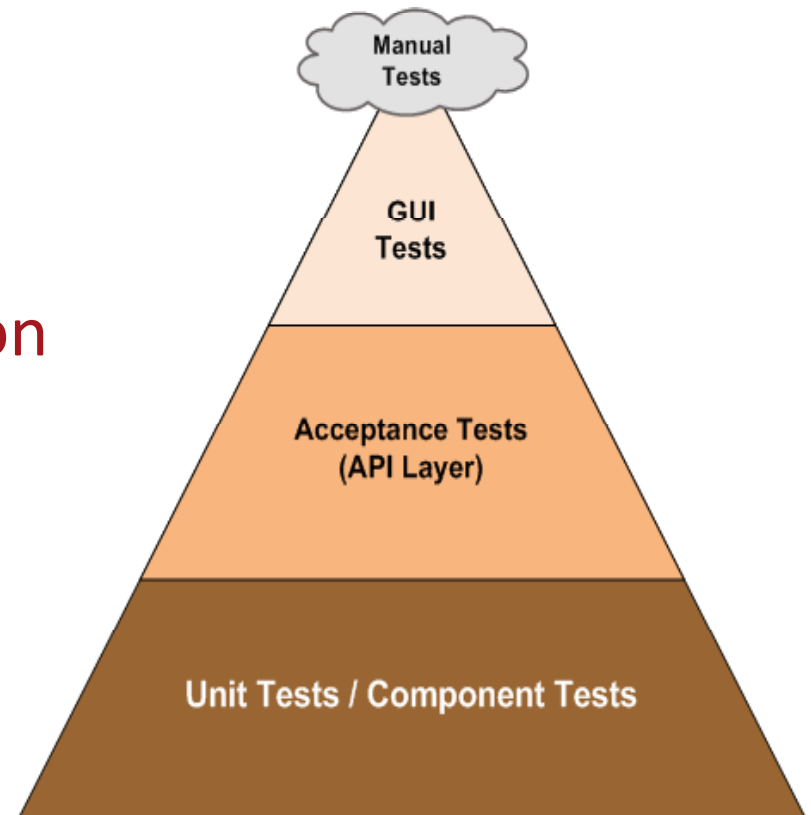


Let's talk a little  
testing ....

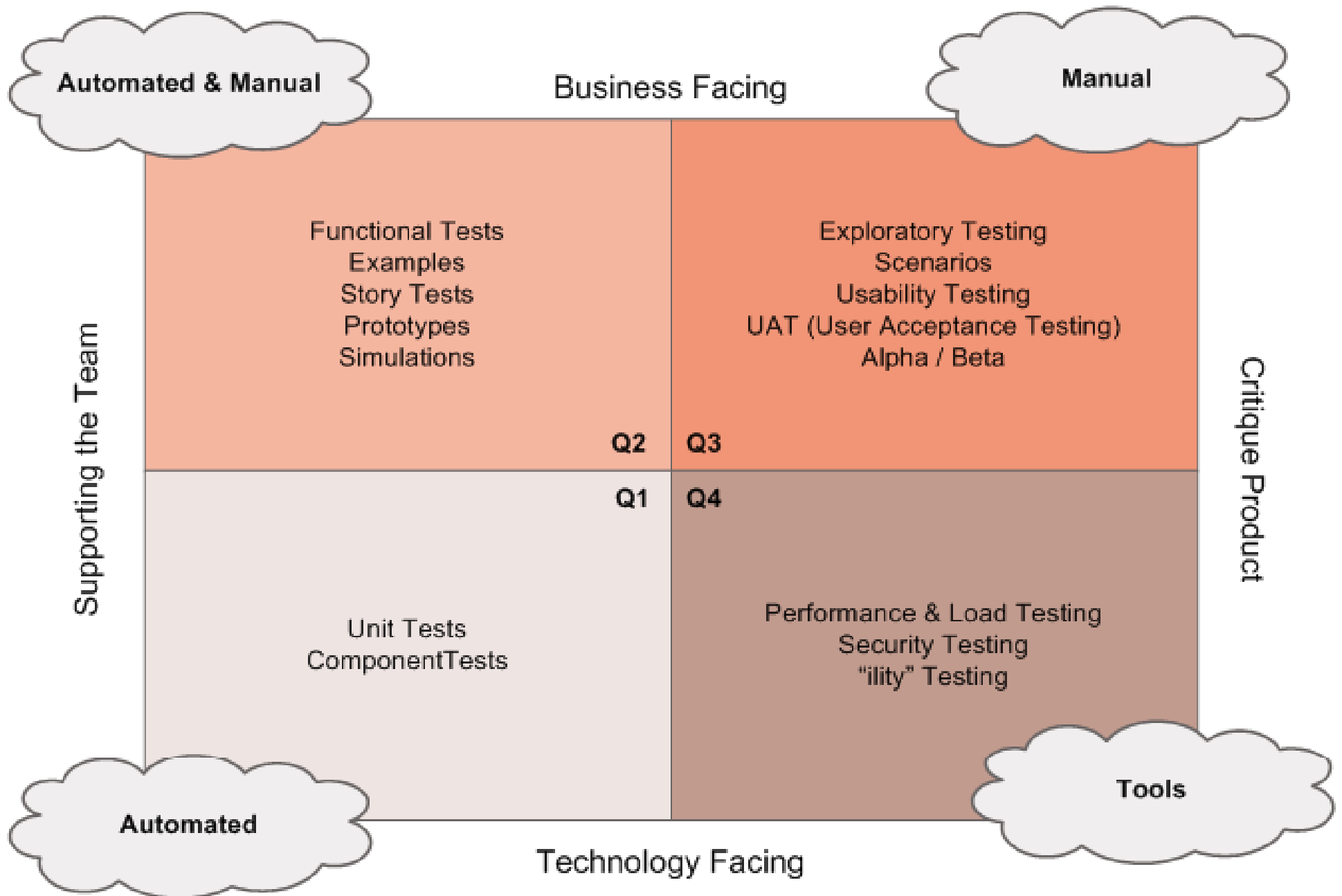


# Automated Test Pyramid - Agile (Mike Cohn)

- Unit tests are the base layer
  - Provide the fastest feedback
  - Best ROI
- Middle layer
  - becomes functional regression tests
- GUI layer
  - May be partially automated
  - Mostly exploratory testing



# The Agile Testing Quadrants (Brian Marick)



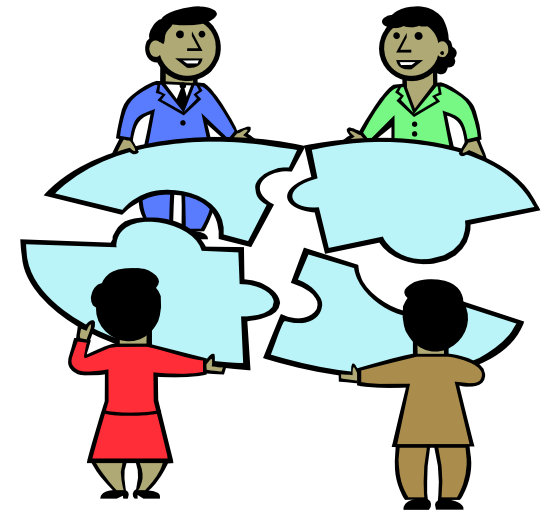
# Use to define 'Doneness'

For each story / feature

- No story is done until tested
- Customer needs captured as passing tests
- Automated regression tests

For release readiness

- Delivers value
- "Doneness" in all quadrants



# Collaborate

---

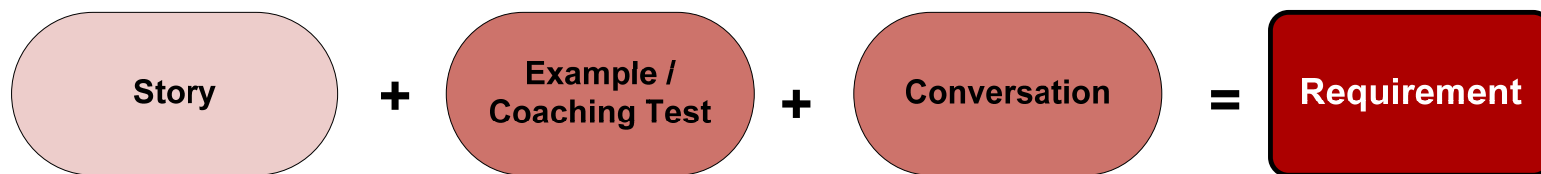
- Collaboration means ...
  - working together
- Communication means
  - sharing ideas, information, decisions & solutions
- Tools for collaboration
  - Feedback
  - Interaction
  - Visibility - allows people to talk about it.



# 'Tester' Activities

---

- Find hidden assumptions
- Ask “why?” Ask “what if?”
- Define and estimate testing tasks
- Ask what other tasks are required
- Define high level story tests or examples



+



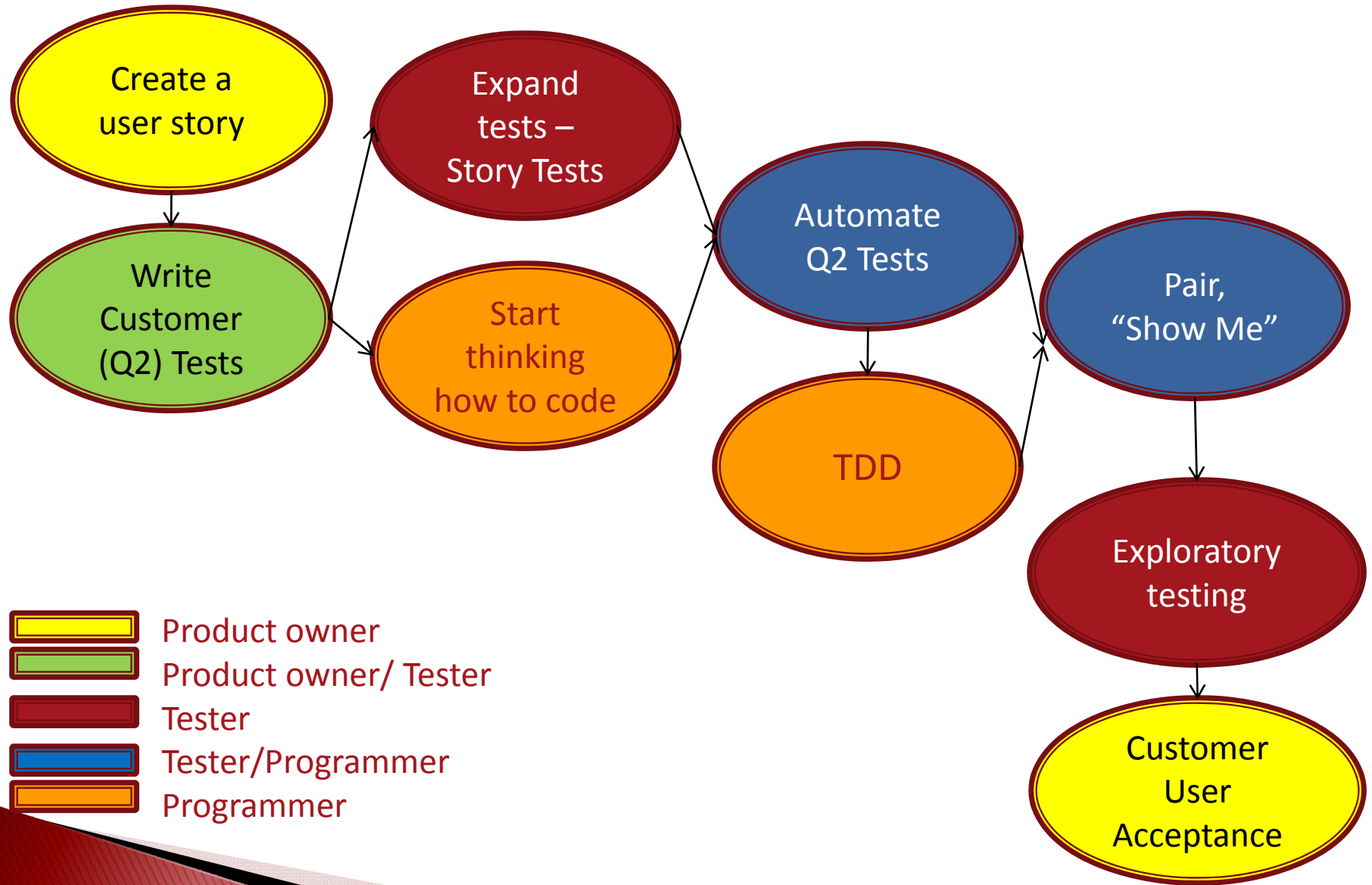
# ATDD – Acceptance Test Driven Dev.

---

- Think Quadrant 2 tests
- Express the intent of the story
- Use examples as specific instances of a scenario
- Think expected and unexpected behaviors
- Gives a shared common understanding of the story
- Feeds into TDD (Test Driven Development)



# ATDD (Acceptance Test Driven Dev)



# Example Story

As a new user, I want to create an account with a user name and password so that only I can access my information.

=====



## Acceptance Tests – Sentence Format

---

- I enter a valid user name and password and am logged into the system and on the home page.
- If I enter an invalid user name, I get an error message “Invalid User Name”, and I am able to try again.
- If I enter an invalid password, I get an error message “Invalid Password”, and I am able to try again.



# Acceptance Test – BDD Style

---

BDD – Behavioural Driven Development

**Given** the user has no existing account

**When** she requests to create a new account,

**Then** she is presented with a screen to enter a valid user name and valid password (rules defined)

**And** the information is saved upon submitting.



# Acceptance Tests – Tabular Style

User Name	Password	Expected result	comments
JanetGregory	Password	Login	Valid combo saved
Janet Gregory	Password	Error	Space in user name
JanetGregory	Abc	Error	Invalid password



# When Coding Starts

---

- Write detailed test cases from examples
  - Make them executable if possible
- Write simple, happy path tests to automate
  - Share them with the programmers
- Once simple tests pass, write more test cases
  - Start with happy path
  - Then move towards edge cases
- Identify possible exploratory scenarios



# Expanded Tests Examples – To Automate

User Name	Password	Expected result	comments
JanetGregory	Password	Login	Valid combo saved
Janet Gregory	Password	Error	Space in user name
JanetGregory	Abc	Error	Invalid password
Janet#Gregory	Password	Error	Special char no allowed
	Password	Error	Blank user name
JanetGregory		Error	Blank password
JanetGregory	Password	Error	User already exists

# Testing/Coding: One Process

---

- Is any testable part of a story ready?
  - Test with behind-the-GUI tool such as FIT?
  - Or other harness to bypass GUI
- Pair with programmers
- “Show Me”
  - Test together before check-in
  - Show them issues
  - Ask questions
  - Bugs found here are cheap and easy to fix



# Exploratory Testing (after coding is complete..)

---

(Cem Kaner, James Bach, Jon Bach, Jonathan Kohl, Elisabeth Hendrickson)

- Simultaneous learning, test design and test execution
- Apply heuristics and techniques in a disciplined way
- Doing reveals more implications than thinking
- Like putting together a jigsaw puzzle
- Use session based testing to
  - Time box your testing
  - Focus your efforts
  - Record results



# System Testing

---

- Business Users and Testers
  - Final manual GUI testing
  - Final regression tests (automated & manual)
- In some teams, the developers run
  - Load & performance tests
  - Stress tests
- Sometimes an implementation team
  - Tests installation or upgrade scripts



# Retrospective – Continuous Improvement

---

- Solve problems as a team
- Is there one single limiting factor?
  - something holding you back
  - something causing a blown iteration
- Action items
  - Focus on 1 or 2 things to improve
  - Create task cards for future iteration



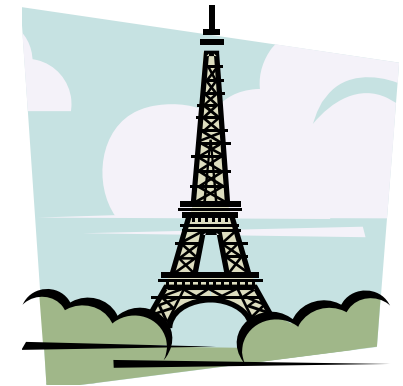
# Remember .....

- Testers are NOT responsible for quality
  - The whole team is
- Programmers do not code alone
  - Everyone helps them understand what to code
- Your team needs the “right” roles and people
  - Those in transition may feel protective of role
  - New skills may be needed



# Whole Team = Key Success Factor

- Can't "test quality in"
- Whole team solves problems
- Whole team thinks testing
- Everyone collaborates
- Team is committed to quality



# Now Available

## *Agile Testing: A Practical Guide for Testers and Agile Teams*

By Lisa Crispin and Janet Gregory

[www.agiletester.ca](http://www.agiletester.ca)

My contact info

[www.janetgregory.ca](http://www.janetgregory.ca)

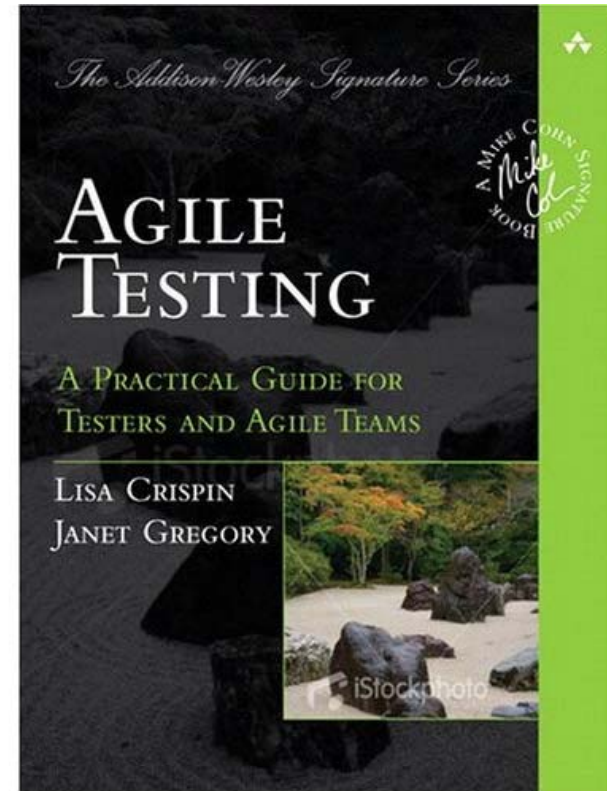
<http://janetgregory.blogspot.com/>

Email: [janet@agiletester.ca](mailto:janet@agiletester.ca)

[www.lisacrispin.com](http://www.lisacrispin.com)

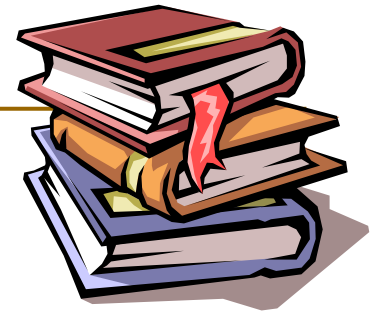
<http://lisacrispin.com>

Email: [lisa@agiletester.ca](mailto:lisa@agiletester.ca)



# Resources

---



- [www.lisacrispin.com](http://www.lisacrispin.com)
- [agile-testing@yahoogroups.com](mailto:agile-testing@yahoogroups.com)
- [www.testobsessed.com](http://www.testobsessed.com) (heuristics cheat sheet)
- Mary Poppendieck and Tom Poppendieck, *Lean Software Development*, Addison-Wesley (series of 3)
- [www.mountangoatsoftware.com](http://www.mountangoatsoftware.com) – Mike Cohn’s web site
- <http://www.exampler.com> - Brian Marick’s web site
- Gojko Adzic, *Bridging the Communication Gap*, 2009
- [www.stickyminds.com/](http://www.stickyminds.com/) Pragmatic Personas - Jeff Patton’s weekly column 1/25/2010
- Jean Tabaka, *Collaboration Explained*, 2006 Addison-Wesley
- Ester Derby and Dianne Larsen, *Agile Retrospectives*
- [http://www.uie.com/articles/indispensable\\_skills](http://www.uie.com/articles/indispensable_skills)
- Agile Manifesto: <http://agilemanifesto.org/>
- [www.satisfice.com/articles/sbtm.pdf](http://www.satisfice.com/articles/sbtm.pdf)
- [www.developsense.com/2009/04/of-testing-tours-and-dashboards.html](http://www.developsense.com/2009/04/of-testing-tours-and-dashboards.html)
- James Whittaker, *Exploratory Software Testing*, 2009 Addison-Wesley

