Coding Dojo: Designing Test Cases for the Gilded Rose Kata

October 2016

Peter Kofler, 'Code Cop'
@codecopkofler
www.code-cop.org

Peter Kofler

- Ph.D. (Appl. Math.)

- Professional Software Developer for 15+ years

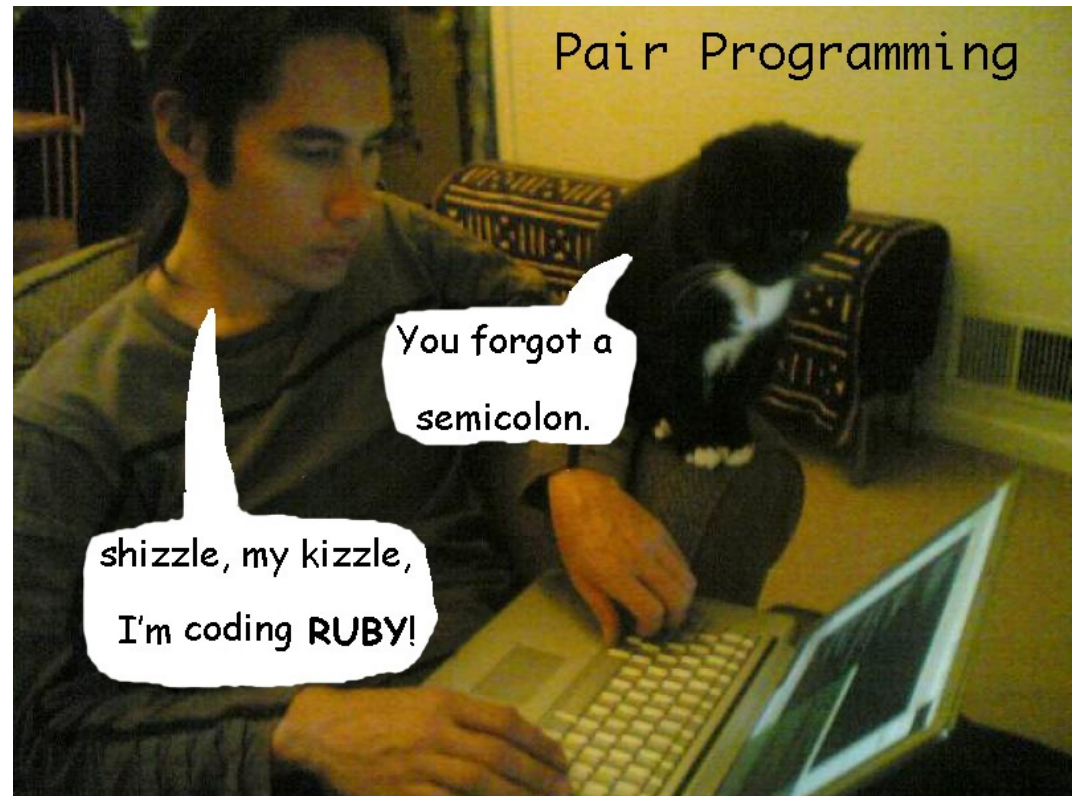- "fanatic about code quality"

- I help development teams

I help development teams with

- Professionalism

- Quality and Productivity

- Continuous Improvement

Mentoring

- Pair Programming

- Programming Workshops

- Deliberate Practice, e.g. Coding Dojos

Developing Quality Software Developers

Coding Dojo? Expectations?

Coding Dojo Mindset

- Safe place outside work

- We are here to learn

- Need to slow down

- Focus on doing it right

- Collaborative Game

Dojo Structure

- Introduction 15'
- Coding 60'
- Retrospective 15'
- Break 15'
- Coding 60'
- Retrospective 15'
- etc.

Clean, Readable and Expressive Unit Test Cases?

(Buzzword Alarm ;-)

Readable?

- **Easily** read; **legible**:
  e.g. a readable typeface.
- **Enjoyable** or **interesting** to read:
  e.g. a readable story.

- What if a test suite would be a readable document at the same time?

Expressive?

- Full of expression; **meaningful**:
  e.g. an expressive shrug.
- **Effectively** conveying **thought**:
  e.g. an expressive glance.

- What if tests revealed their intend?
  Would express what should happen?

Clean?

- **Free from dirt** or marks:
  e.g. a clean kitchen floor.
- **Without imperfections** or errors:
  e.g. a clean edge.

- What if all your tests would be nicely structured and consistent?

TDD Skills

- Driving Development with Tests
- Designing Test Cases ⬅
- Designing Clean Code
- Refactoring Safely

Welcome to team Gilded Rose

The existing inventory system

- We have **items** to sell. Items degrade in quality the older they get.

- All items have a **SellIn value** which denotes the number of days we have to sell the item.

- All items have a **Quality value** which denotes how valuable the item is.

Requirements

- At the end of each day our system lowers both values for every item.

- Once the sell by date has passed, Quality degrades twice as fast.

- The Quality of an item is never negative.

- The Quality is never more than 50.

Special Item: Brie

- *Aged Brie* actually increases in Quality the older it gets.

Backstage Passes

- *A backstage pass* increases in Quality as it's SellIn value approaches (by a complex formula)

- but Quality drops to 0 after the concert.

Special Item

- *Sulfuras*, a legendary item, never has to be sold or decreases in Quality.

Assignment

- Get the code

- Run tests, should see a failing test

- Read *GildedRoseRequirements.txt*

- Create "perfect" unit tests

  - derive test cases from requirements
  - cover all cases e.g. boundary conditions
  - readable, concise, free of duplication

Create a test suite that is a readable document at the same time!
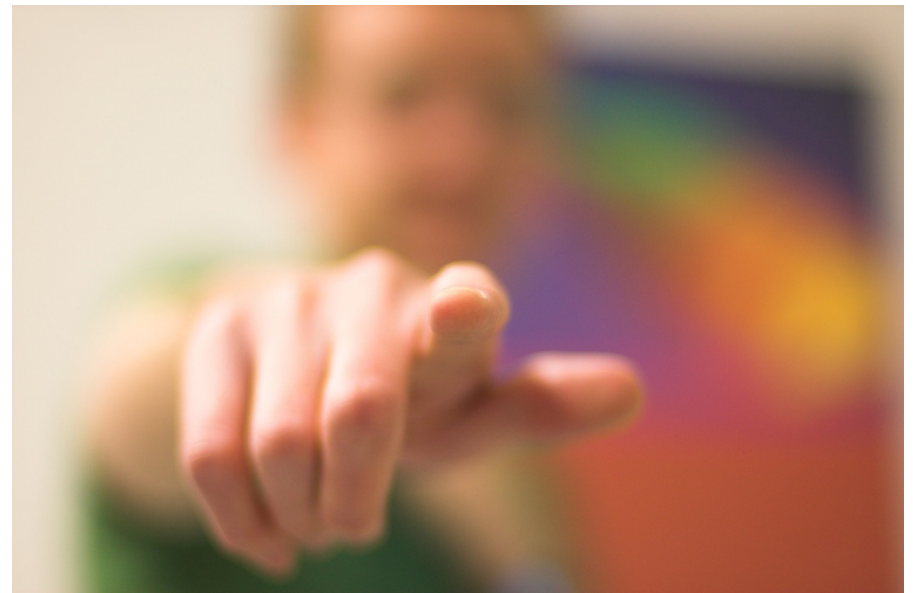
What Will We Do Now?

- Derive test cases from requirements.

- Find all necessary test cases.

- Create "perfect" tests.

- Name test cases accordingly.

- Test boundary conditions.

- Readable, concise, free of duplication.

- Experiment with styles.

Don't Focus on Getting it Done. Focus on Doing It Perfectly.

$\rightarrow$ Practice

Closing Circle

• What did you learn today?

• What surprised you today?

• What will you do differently in the future?

Peter Kofler

@codecopkofler

www.code-cop.org

Kata by

Emily Bache

@emilybache

http://coding-is-like-cooking.info/2013/03/writing-good-tests-for-the-gilded-rose-kata/

CC Images

- Bruce http://www.flickr.com/photos/sherpas428/4350620602/

- pairing http://www.flickr.com/photos/dav/94735395/

- dojo http://www.flickr.com/photos/49715404@N00/3267627038/

- agenda http://www.flickr.com/photos/24293932@N00/2752221871/

- inn http://www.flickr.com/photos/danielleblue/170496395/

- Brie http://www.flickr.com/photos/chez_loulou/2767503201

- Pass http://www.flickr.com/photos/frf_kmeron/5556518514

- Sulfuras https://www.flickr.com/photos/sharelabs/11195626116

- wants you http://www.flickr.com/photos/shutter/105497713/