

# Software Testing and XP

*Developing with quality in mind?*

Tim Bacon, ThoughtWorks Ltd  
[tbacon@thoughtworks.com](mailto:tbacon@thoughtworks.com)

## Why test?

---

As developers we write tests...

- ...to tell us when we to move on to the next task
- ...to guide our design and allow it to evolve
- ...to understand when our metaphor no longer applies
- ...to remind us of what we were doing when we were distracted
- ...to see the impact of new code on an existing code base
- ...so that we aren't ignorant when things suddenly break
- ...so that we have "working software" at all times
- ...so that we can code / build & integrate "little and often"
- ...so that we have the courage to mend "broken windows"
- ...so that we have an executable specification of the system
- ...so that we have a meaningful measure of progress

## Testing fits right in with XP

### Practices

- Test driven development
- Pair programming
- Continuous integration
- Planning game
- Small releases
- Domain focus / metaphor
- Simple design
- Refactoring
- Collective ownership
- Sustainable Pace
- On-site Customer
- Coding standard

### Values

- Communication
- Feedback
- Courage
- Simplicity

## Audiences for tests

---

- Developer tests (for & by developers)
  - Guide and document design-in-the-small
  - Verify algorithmic correctness
  - Assert behaviour at class / method level
  - Detect change
- Customer tests (for & by customers)
  - Guide and document design-in-the-large
  - Verify system correctness
  - Assert behaviour at package / deployment level
  - Detect change
  - Measure progress
- NB: people driven “exploratory tests” are still required

## Tool support

---

- Developer tests:
  - Junit <http://junit.org/> (also NUnit, PyUnit, ...)
- Customer tests:
  - Driven from user stories (before code!)
  - FIT <http://fit.c2.com/>
  - Or roll your own test framework
    - *Domain specific / situated test “language”*
    - *Extend existing test runners e.g. for Ant support*
- All tests run as steps in the team build process
  - Developer tests must be “green” always
  - Customer tests can start “red” but must become “green” and then remain that way thereafter

## Questions?

- Further reading:
  - “Test Driven Development”  
ISBN 0321146530
  - “Testing Extreme Programming”  
ISBN 0321113551
  - “Lessons Learned in Software Testing”  
ISBN 0471081124
  - “Better Software” magazine
  - Blogs: Brett Pettichord, Brian Marick, James Bach, ...

