Challenges of Training Software Testers

Jesse Rosenthal
ASQ CMQ/OE, CSQE
16 JAN 2008

Assumptions

- Everything is open to debate and discussion!

My Background

- Coordinate (and have managed/conducted) software testing for Department of Defense Military Health System (MHS) Resources Information Technology Program Office (RITPO) since 2003
- Project Management course development and instructor
- Division Lead for Capability Maturity Model (CMM) level 2/3 implementation
- Associate Referee Instructor for United States Soccer Federation

Why test? Customer expectations

- Decrease risk
- Decrease uncertainty
- Increase knowledge
- Confirm expected results can be obtained by users in addition to the developer

Testing Practical Exercise

- Task: Begin a software system integration test in 1 month.

- Industry:
  - Pharmacy
  - Banking
  - Insurance

Internal vs. External Test Teams

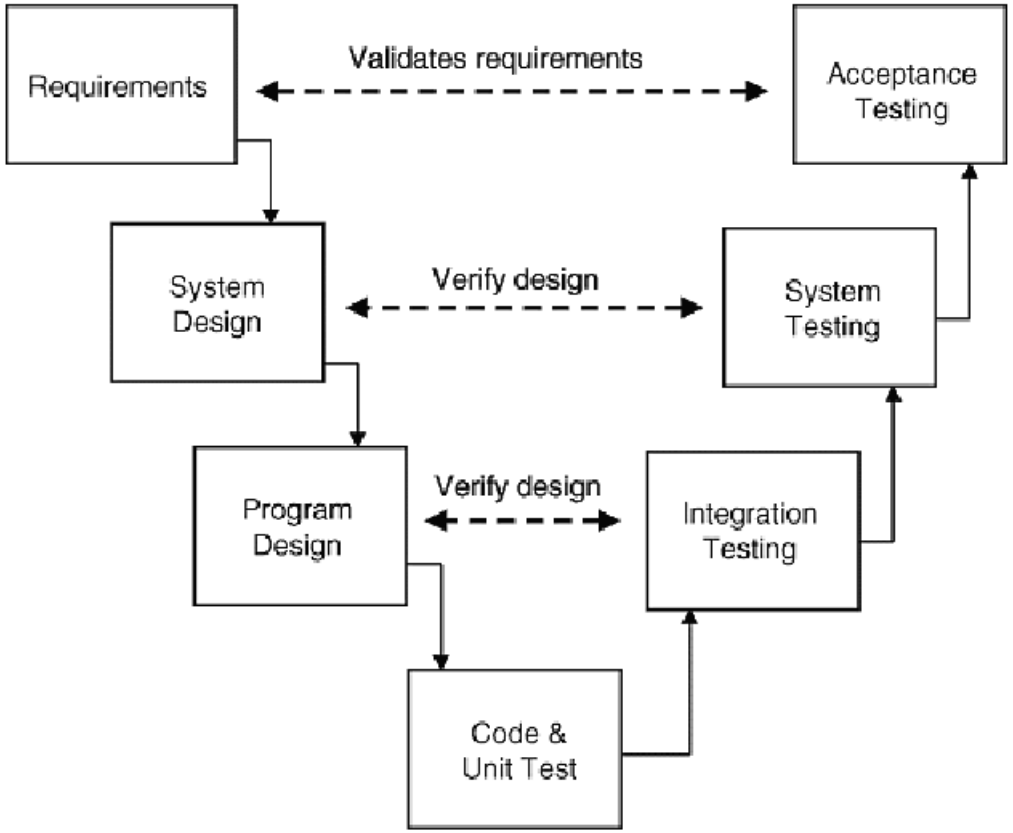
- Internal Test Team
- Ready access to developers to clarify
- Temptation to reuse developers in testing role
- Nominally lower accounting costs

External Test Team

•Easier to preserve independence of testing, validity

•Higher accounting costs, if not necessarily economic costs

Where does testing meet design?



Testing starts with Good Requirements …

- But what makes a requirement good?

Attributes of Good Requirements

- Singular condition to evaluate
- Shall statements
- Unambiguous

NO WHITE ELEPHANTS!

- San Diego Zoo -

Attributes of GOOD Testing

- Provides information about capabilities of system under test.
- Follows documented plans & procedures
- Evaluates whether each approved requirement has been demonstrated
- Evaluates whether the system is robust
- Thorough documentation of issues where expected results do not match actual results
- Evaluates whether the system under test provides graceful error-handling

Attributes of GOOD Testing

- Takes place against defined software baseline
- Takes place in controlled software environment
- Performed by independent evaluator

- Not all testing is good

Attributes of BAD testing

- Results are not documented
- Actual results of testing are not compared to expected results

- With an unlimited testing budget …

But seriously …

- We can only hire one tester to add to the test team. Given the background of the following 4 candidates, who can contribute to our testing success?

Candidates: Preliminary Ranking

- ___) Quality Assurance specialist who is relocating from Milwaukee. Having substantial shop floor experience inside a tire factory performing lot inspections, HR says that he taught Deming everything he knew – or at least that's the impression from the phone screen.

- ___) Community banker who says that she has seen enough HUD-1 statements and consumer credit reports to last a lifetime. Wants to try something else.

Candidates: Preliminary Ranking

- ___) Pharmacist with 2 years experience who scoured the Web looking for the company who tested the last software product installed in his pharmacy. Intends to vent at first opportunity (read: interview panel) to explain how poor their existing testers are and that a trained monkey could take a break from organ grinding and do a better job.

- ___) Software tester with 10 years of experience in aviation software evaluation.

Who to Hire?

- Detail-oriented
- Inquisitive
- “Missouri mentality”
- Open-minded
- Honest
- Responsible

Hiring a tester makes sense …

- … but should we train the tester?
- Or see what happens tabula rasa?

If I don't train the testers, what are the savings?

- Schedule
- Budget

If I DO train the testers …

- Are there benefits?

Should the developer want to provide product training to the test team?

- Dry-run for the trainer in classroom model
- Allows controlled, professional assessment of strengths and weaknesses of training approach
- Allow development of FAQ list for eventual deployment target audience
- Gaps in training will be readily identified and addressed before Help Desk activation

Should independent test manager deploy testers to developer training?

- Provides focus on real issues, not misunderstandings driven by unfamiliarity with the system
- Decrease ramp up time to develop and execute scripts
- Limit inefficiencies by not forcing the testers into trial-and-error mode, expand automated testing scope with more sophisticated product understanding
- Identify risks to success with deployment of system under test with training and documentation

Aside from product training … What training does a software tester need?

- Test Tools
- Test Processes and Procedures
- Issue Documentation Procedures
- Communication protocols

•Industry knowledge/context to understand what requirements capture

Requirements management?

Transition – Hiring to Testing

- We will now assume that each of you has hired the right tester for your team.
- Now what?

Tester Communication Expectations

- Fact-driven
- Fact-limited
- Clarity

Traceability

- Mapping Requirements to Testing
- Mapping Training to Requirements
- Mapping Manual/Exploratory Testing to Legacy Training/Industry Knowledge

- Should we use customers as beta testers?

Tester Incentives – Separate Good from Bad

- Tie compensation to number of issues detected?
- Tie compensation to number of test scripts/cases executed?
- Tie compensation to completion of testing on schedule?
- Tie compensation to completion of testing within budget?

Satisfy customer expectations

- Has testing reduced the risks of deployment?
- Has testing increased confidence in expected results?

Adding more testers …

- If test team has been given training already, do you need to freeze hiring?

- No –you can measure knowledge retention in a trained tester by asking a legacy tester to present training to the new team member

References

- Checkpoint Software White Paper: “Building an Effective Software Testing Practice” by Ken Arneson. 2 JAN 2008
- Borland Software White Paper: “The Top 10 Blunders in Integration and Testing … and How to Avoid Them” by Rob Cheng, Burke Cox, and John Minnihan. March 2007
- LogiGear Corporation White Paper: “Testing Under Pressure -Relieving the "Crunch Zone” by Hans Buwalda

Conclusions