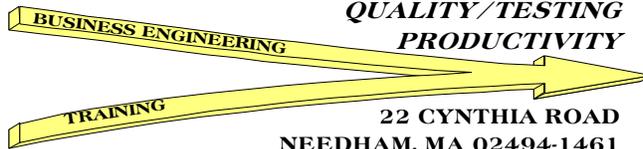


How to Get QA Meaningfully Involved Earlier



Robin F. Goldsmith, JD
GO PRO MANAGEMENT, INC.
 SYSTEM ACQUISITION & DEVELOPMENT
 QUALITY/TESTING
 PRODUCTIVITY



22 CYNTHIA ROAD
 NEEDHAM, MA 02494-1461
 INFO@GOPROMANAGEMENT.COM
 WWW.GOPROMANAGEMENT.COM
 (781) 444-5753

Impact of Projects Involving QA Late (after code has been written)

| Projects | No. | Test Adequacy | \$ Impact of Prod Defects | | | | |
|----------|-----|---------------|---------------------------|----------|-----------------|----------|----------|
| | | | Hi Prod Defects | | Lo Prod Defects | | |
| 0-49% | ___ | OK | ___ | Hi\$ ___ | Lo\$ ___ | Hi\$ ___ | Lo\$ ___ |
| | | No | ___ | Hi\$ ___ | Lo\$ ___ | Hi\$ ___ | Lo\$ ___ |
| 50-75% | ___ | OK | ___ | Hi\$ ___ | Lo\$ ___ | Hi\$ ___ | Lo\$ ___ |
| | | No | ___ | Hi\$ ___ | Lo\$ ___ | Hi\$ ___ | Lo\$ ___ |
| 76-100% | ___ | OK | ___ | Hi\$ ___ | Lo\$ ___ | Hi\$ ___ | Lo\$ ___ |
| | | No | ___ | Hi\$ ___ | Lo\$ ___ | Hi\$ ___ | Lo\$ ___ |

//// *Ways You'd Like QA Involved Earlier*

Technique

Why? Benefit

-
-
-
-
-
-
-
-

//// *Reasons for Not Involving QA Earlier*

-

-

■■■■ Conventional Wisdom

- Late QA involvement means not enough time to
 - Design and create sufficient tests
 - Execute even the tests we've had time to create
 - Fix some of the defects that are found
- Earlier QA involvement enables
 - Designing and creating more, more thorough tests
 - Executing more of the tests because no delay
 - *Maybe having users review test cases*
 - *Detecting untestable requirements in time to fix them*

■■■■ Unconventional Wisdom

- How would earlier QA involvement *by itself* actually result in appreciably
 - Designing and creating more thorough tests?
 - Fixing more of the (even greater number of) detected defects? ***Marginal or Material concerns?***
- Conventional methods focused on tail-end tests and even on detecting untestable requirements do little to identify bigger issue: *design errors due to wrong and overlooked requirements*

■ ■ ■ ■ Doesn't Agile Eliminate the QA Involvement Issue? Maybe, but ...

- Product Owner active participation identifies and prioritizes requirements/user stories to work on
- And user story acceptance criteria to test
- Developers define and automate test-first (unit) tests

Is eliminating QA actually the right direction?

■ ■ ■ ■ Why Agile Still Needs QA--and Early

-
-

Conventional and Agile Both Tend to Jump into Detail

- Test cases represent small risks
 - Focus on them makes it likely to miss bigger risks
 - Users have little interest in or patience reviewing test-case-sized small risks
- User stories fall down in integrations
 - Almost sole focus on individual backlog items
 - Intentionally ignores design of bigger pieces
 - User story acceptance tests don't address either

Instead, Apply Proactive Testing™

- More than 21 ways to review requirements
 - Buy-In → – Content—wrong and overlooked, *major value*
 - Format—clarity and testability, *less value*
 - More than 15 ways to review designs—ditto
 - Identify large risks first
 - Buy-In → – Involve stakeholders in identifying
 - Backlog → – Prioritize and drive down to medium-size risks
 - Prioritize and drive down to small risk test cases
- Write low-overhead test cases: inputs, expected results**
Supplement and refine with exploratory