



QAI /QAAM 2011 Conference
"Proven Practices For Managing
and Testing IT Projects"



Writing Testable Requirements

Bill Rinko-Gay

September 27, 2011



Agenda

- Good Requirements
- Testable Requirements
- Roles
- Verifiable Requirements
- Verifying Requirements
- Recap
- Q&A



QA – The Goal

Good Requirements

- What makes requirements good?
 - Correct
 - Complete
 - Clear
 - Testable

Testable Requirements

- Decomposed
 - The requirement is broken down into its smallest constituent part
- Measurable
 - Objective measure
 - Measure clearly relates to requirement
- Unambiguous
 - Clear understanding
 - Only one right answer

Roles

- In your organization, who writes requirements?
- Why is this a subject at a QA conference?
- Can we agree ?
 - QA is responsible for the quality of the outputs of all phases of the SDLC
 - QA pays the price for failure in any phase of the SDLC
- BA is responsible for correctness
- QA can use experience and common sense to look for traps

Verifiable Requirements

- Complete
 - Customer and BA will often focus on functional requirements
 - QA ensures non-functional requirements are considered
 - Maintainability
 - Supportability
 - Scalability
 - Robustness
 - Performance

Verifiable Requirements

- Clear
 - Clear language is hard
 - Active voice
 - “The system shall...”
 - Free of jargon / buzzwords
 - Free of “fluff”
 - Unambiguous



Ambiguity

- “Green Field” example

Verifying Requirements

- Compare your “green field” vision with that of your development team
 - Compare your Test Design with the Code Design
 - Test Plan – roles, resources, risks (Project Plan)
 - Test Design – how you will accomplish the test (Code Design)
 - Test Cases – the actual steps (Code)

Verifying Requirements

- Why a Test Design?
 - Describes your approach in terms anyone can understand
 - Maintains the right level – does not need to be re-written unless the Code Design needs to be re-written
 - Test plan does not include the approach
 - Test cases are too detailed
 - Most people don't have the patience or skill to review
 - Need to be re-written if the code implementation changes

Verifying Requirements

- Test Design Template
 - Function / Feature / Requirement to be tested (from RTM)
 - Functional Test Cases
 - Negative Test Cases
 - Non-functional Test Cases
 - Special data requirements
 - Special environment requirements
 - System Testing
 - Integration Test Cases
 - Non-functional Test Cases
 - Special data requirements
 - Special environment requirements

Recap

- QA does not write requirements, but QA pays if the requirements are not testable
- Testable requirements are unambiguous and measurable
- Test Design is the tool to ensure requirements are unambiguous
 - Additional benefit of a good planning tool that makes writing test cases easier



Q&A