Chapter 10

Testing and Change Management

Copyright © 2000 by Addison Wesley
Version 1.0

Topics

- Testing System Services
- Testing System Constraints
  - Test and Change Management Documents
  - Managing Change
- Traceability
Testing system services

- **Informal testing**
- **Methodical testing**
- **Nonexecution-based (formal reviews)**
  - Walkthroughs
  - Inspections
- **Execution-based**
  - Testing to specs
  - Testing to code

Walkthrough

- **Formal review meeting** but of a brainstorm flavor
- **Can be conducted**
  - In any development phase
  - On a regular basis (e.g. weekly)
- The **moderator** organizes and conducts the meeting
- **Participants** prepare prior to the meeting
- The **purpose** of the meeting is to identify and confirm the existence of a problem (defect), not to attempt to solve it
**Inspection**

- Similar to walkthrough but includes project managers
  - Like walkthrough
    - To identify defects, not to solve them
  - Unlike walkthrough
    - Conducted less frequently
    - Targets selected and critical issues
    - More formal and rigorous
    - Organized in a number of stages
      - Planning
      - Informational meeting
      - Meeting
      - Reinspection

---

**Testing to specs**

- Execution-based testing
  - Applies to software, not documents or models
- Black-box testing
- Refers to test requirements
  - Test case document - test scenario for the tester
  - Can be recorded in capture-playback tool
    - Regression testing
- Discovers defects difficult to find with testing to code
  - Missing functionality
- Cannot be exhaustive
Testing to code

- Execution-based
- White-box (glass-box) testing
- Exercises the code
- Can be
  - Supported by capture-playback tools
    - Many scripts must be written by the programmer rather than generated
  - Used for regression testing
- Cannot be exhaustive

Testing system constraints

- Predominantly execution-based
- Includes
  - User interface testing
  - Database testing
  - Authorization testing
  - Performance testing
  - Stress testing
  - Failover testing
  - Configuration testing
  - Installation testing
User interface testing

- Starts in requirements phase
  - Storyboarding
  - Windows in use case documents
  - GUI prototypes
- Post-implementation GUI testing
  - Developers
  - Testers
  - Customers (pilot tests)
- Like many other kinds of tests, requires answering a wide range of questions and marking each question as
  - Passed
  - Passed conditionally (explanation required)
  - Failed

Database testing

- Inherent in many other kinds of tests
- Done as
  - Black-box and
  - White-box
- Tests needed for
  - Transaction processing
  - Performance
  - Concurrency
  - Authorization
- Some DB tests can be included as part of other tests
  - E.g. performance test would include DB performance
Authorization testing

- To verify the security mechanisms
  - In the Client and
  - In the Server
- Authentication
- Authorization
  - GUI to configure dynamically to authorization level of the user
  - Server permissions (privileges) granted to the user
    - To access server objects
    - To execute SQL statements
- Authorization design
  - User level
  - Group level
  - Role level

Other tests

- Performance testing
  - Transaction speed and throughput
  - Peak loads
- Stress testing
  - For abnormal demands
- Failover testing
  - For hardware, network and software malfunctions
- Configuration testing
  - For various software and hardware configurations
- Installation testing
  - For every installation platform
Test and change management documents

Business Use Case Document
- Feature 1
- Feature 2
- ...

Use Case Documents
- Use Case Req 1
- Use Case Req 2
- ...

Enhancements Document
- Enhancement 1
- Enhancement 2
- ...

Test Plan Document
- Test Case 1
- Test Case 2
- ...

Test Case Documents
- Test Req 1
- Test Req 2
- ...

Defects Document
- Defect 1
- Defect 2
- ...

Enhancements Document
- Enhancement 1
- Enhancement 2
- ...

Test case document

MACIASZEK (2001): Req Analysis & Syst Design
Managing change

- **Testing → defect**
- **Defect → change request**
  - To be fixed
  - Enhancement
- **Defects and enhancements**
  - Undergo status changes
  - May be prioritized
  - Have owners
  - Are traced back to test and use case documents
- **Managing change in multi-developer projects**
  - Locking
  - Version control

Submitting change request

- **Submitting**
  - Entering defect or enhancement into project repository
  - Submitted status
  - Can result in automatic email notifications to interested team members
- **Possible actions on submitted change request**
  - Assign (to team member)
  - Modify (the change request)
  - Postpone
  - Delete (without fixing)
  - Close (probably after fixing)
Keeping track of change requests

- Assigned to a team member
- Only one team member can open it at a time
- Open state → Resolve state
  - Email notifications
- Resolve state → Verify state
- Charts and reports (project metrics)
  - Number of unassigned defects
  - Allocations to team members
  - Number of unresolved defects

Traceability

- Cost-benefit analysis for traceability
  - Traceability requires time and money
  - Projects can fail if change not traced and managed
  - Long term benefit – project management with metrics
- Levels of traceability
  - Use case requirements → defects
  - Use case requirements → test requirements → defects
  - Also system features, test verification points, enhancements, etc.
System features → use cases → use case requirements

Test plans → test cases → test requirements
Use case → document

Use case → requirement
Use case requirements → test requirements

Test requirements → defects
Use case requirements → enhancements

- Defects are traced back to test requirements
- Enhancements are traced back to use case requirements
- Enhancements can also be traced back to defects (when a minor defect has been postponed to a future release of the product)

Summary

- Testing divides into
  - testing of system services and
  - testing of system constraints
- **Testing of system services** can be nonexecution-based or execution-based.
- **Testing of system constraints** includes testing of user interface, database, authorization, performance, stress, failover, configuration, and installation
- Testing and change management require specialized **documentation**, such as test plans, test case documents, defect and enhancement documents
- **Change request** is normally either a **defect** or an **enhancement**
- A vital part of the change management tool relates to the establishing of **traceability** paths