#### Software Test

#### Lesson 12 Test Execution v1.0

Uwe Gühl, Jittat Fakcharoenphol



Fall 2007/ 2008

# Contents



- Test Execution
  - Goal
  - Involved people
  - Test Suite
  - Test Environment
  - Release Management
  - Defect Management
  - Regression Tests
  - Reporting
  - Test Stages

#### A sample testing cycle Test Execution





#### Test Execution Goal



- Deliver all necessary informations to make the acceptance of the software possible
  - Software Quality Report
- Helping the project to achieve best quality in software
  - There should be only defects at the end tolerable following the regulation of acceptance



#### Test Execution People

- Test Manager
- Defect Manager
- Tester (Test Engineers)
- Test Data Manager
- Environment Manager
- Collaboration with
  - Specification creators
  - Software developer
  - Operation

#### Test Execution Test Suite



- Test Cases, Test Scenarios, and Test Data have to be updated, overworked, and added because
  - of changes in the specification
  - there are defects and faults in them
  - they were forgotten
  - there are more needed to test specific areas of the software more detailed

#### Test Execution Test Environment



- Goal: Simulation of regular operation
  - System environment
  - Operation system
  - Interfaces Simulators of participating systems or their test version
  - Hardware (for example cable capacity)

## Test Execution Release Management



- Release plan
  - Optimal: Short installation time (weekend), to use resources optimal (it is costly if testers could not test, because the system is not available)
- Name convention Example
  - <Release>.<Wave>.<Patch>.<Hot fix>
  - Software version 1.2.0.0
     (Release 1, Wave 2, Patch 0, Hot fix 0)
- Smoke test

#### Test Execution Release Management



• Example for Release plan

Planned EDC	Release /Patch/ Hot		
Deployment	Fix #	Content	
27/11/2006	1.2.0.0	Release of SR1 Wave 2 - 64 Use Cases	
04/12/2006	1.2.1.0	Patch for Defects resolved in past week	
12.11.2006	1.2.2.0	Patch for 4 Use Cases and 57 Defects resolved in past week	
12.12.2006	1.2.2.1	Hot Fix for Data	
13/12/2006	1.2.2.2	Hot Fix for Defect 596	
18/12/2006	1.2.3.0	Patch with 12 Use Cases, Incremental Test Data, and Defects	
		resolved during past week	
18/12/2006	1.2.3.1	Hot Fix for Client code along with DB scripts	
20/12/2006	1.2.3.2	Hot Fix for smoke test bugs during application matrix testing	
27/12/2006	1.2.4.0	Patch with Defects resolved during past week	
05/01/2007	1.2.5.0	Patch with Defects resolved during Christmas break	
05/01/2007	1.2.5.1	Hot Fix for Severity 1 Defect, if necessary	
13/01/2007	1.2.6.0	Patch for Change Request items - Delegation and LDAP and	
		Defects resolved during past week	
23/01/2007	1.2.7.0	Patch for Defects resolved in past week	
24/01/2007	1.2.7.1	Hot Fix for Severity 1 Defect, if necessary	
30/01/2007	1.2.8.0	Patch for Defects resolved in past week	



- How to write reports concerning defects?
  - Describe everything that the defect could be reproduced
  - Try to help that the defect could be fixed
  - Don't joke
  - Write as neutral as possible
  - Describe how the defect occurs step by step
  - Describe what works and describe what went wrong
  - Use screen shots
- If you have an idea, suggest your idea as a proposal
   Jittat, Uwe Software-Test 12 v1.0



X

#### Test Execution Defect Management

- How to write reports? Example
  - Step 10: enter zip code
  - Step 20: do not enter city name
  - Step 30: verify data base entry zip code
  - Step 40: verify data base entry city name Nullpointer exception: Window with unreadable message appears, but could be closed Idea: A check for city name before sending the data to the server.
  - Step 50: Error message displayed

Important: Your task is to report the bug in the best way so it could be fixed – Ideas are <u>really</u> <u>only optional</u>



- Defects and Test Cases
  - Relation is m:n
  - A Test Case could have several defects (*Hint: That's why design small Test Cases*)
  - Problem: If an execution of a Test Case has to be stopped, possible defects in the following test steps could only be detected, if the defect is fixed
  - A defect could block other Test Cases
     Example: Interface tests



- Tasks:
  - Daily communication
  - Defects Coordination with tester, customers, and software vendor (developers)
    - Collection and administration
    - Assigning of severity and priority levels
    - Clarification of responsibilities
  - Monitoring of defect fixing
    - Monitoring releases: Which defects were fixed and delivered?
    - Organize retesting



Example: Execution of Test Cases





Example: Defect Management Process





Example: Defect Management, Status transitions



#### Test Execution Regression Test



- After delivery of a version / release / patch principally everything what have been tested has to be repeated
- Because of possible site effects it could not be guaranteed that every successful passed test would pass again!
  - $\rightarrow$  Test automation



- Main contents of a test report:
  - Test progress
    - Are we in plan? Compare target / actual
  - Test coverage
    - How much of which areas has been tested?
    - How much testing is blocked because of defects?
  - Defects and their status
    - How many defects found? Open / Closed defects
    - Severity / Priority level
  - Statements
    - Confidence in quality of testing
    - Important information: What happened and should be shared?



- Test reports should be
  - reported regularly most suitable weekly, depending on (test) situation towards end daily
  - written exactly, neutral and true
    - do not euphemize, do not overdraw, do not cover up
    - no accusations, no fingerpointing
  - short

A presentation about 3 to 4 pages should be sufficient, more detailed information could be put into the backup



- Suggestion: Act as a pressman
  - Regularly report
    - Put the bug report statistics to a later page, so interested people read first things you would like to address like
      - Requests to the management concerning decisions
      - Information about administrative problems in testing
      - Important critical defects
      - Important statements concerning testing progress
  - Final report
    - Report something like the top 10 issues in the final report so that it's clear what kind of quality the customer could accept



- Test diary: What happened?
  - When were deliveries?
  - When was testing not possible?
  - Could be basic for reports



- Consider Input and Output criteria
- Optional pre phase to establish the processes

recommended



- Hints how to proceed
  - Try to broaden testing as soon as possible for first statements
  - (Certainly) start with high prioritized tests
  - Where You find relatively many defects enforce testing, establish a special testing group (wasps' nest)
  - Objective proceeding:
    - Defects should be found and fixed
    - Don't discuss about reasons and responsibilities
    - For this purpose use for example lessons learned workshops





• Example Test process





<ul> <li>Example Test week</li> </ul>						Operations Tester
					Environment Mgr	
	Tuesday	Wednesday	Thursday	Friday	Weekend	Monday
Test Team is working on test executions and open defects in Tool if necessary						
Software Vendor fixes defects		1				
Test Management and Software Vendor decide which defects will be fixed in which patch.	_					
Environment Manager informs Operations about patch delivery on next Monday						
Operation plan their activities						
Software Vendor delivers release with release notes				▶ 		
<ul> <li>Test Management verifies the release notes (e. g. are there any existing tasks, dependences or effects, should be the delivery earlier)</li> </ul>						
<ul> <li>Environment Manager informs</li> <li>Operations which issue will change in environment</li> </ul>						
Operations gives advance notice for software delivery on Monday						_
Downtime for installation during 9 and 11 a m						
<ul> <li>Information of duration to Test</li> <li>Management and Tester</li> </ul>						
Tester start testing again				1   		



## Test Execution Test Stages - 02-1 ST



#### • Example Entry criteria for System Test

Entry Criteria:	Subject	Responsible	Yes / No		
<b>Preparation</b>	845,000				
<ul> <li>Unit testing a documented</li> </ul>					
I est data for					
<ul> <li>Test environr</li> </ul>					
<ul> <li>All test cases the target tes</li> </ul>					
<ul> <li>Specific inter</li> </ul>	nationalization criteria?				
<u>Defects</u>					
<ul> <li>Severity Leve</li> </ul>	el 1 defects: none				
<ul> <li>Severity Leve workarounds</li> </ul>	el 2 defects: documented, description of correction defined in the release plan				
Severity Level in the release	el 3 defects: documented and correction defined				



#### Test Execution Test Stages - 02-1 ST

- Proceeding
  - Planning of Test Execution following prioritization
  - Allocating Test Cases to Tester
    - following prioritization
    - focusing on critical areas
    - taking care concerning coverage
  - Regular meetings with testers, developers, ... e. g.
    - Morning: What to test, focus, news concerning the system to test, ....
    - Evening: Testing results, developer statements, defect meeting, next steps, ...

#### Test Execution Test Stages - 02-1 ST



- Documentation (typically standardized or supported by a Test Management System) – base for report
  - Test execution reporting on a Test Case
    - Test execution status (passed, failed, blocked, ...)
    - Tester
    - Test execution date, time
    - Tested software version
    - Used hardware / software (operating system)
    - Test execution comments
    - Connection to defects



# Test Execution Test Stages - 02-2 SIT



- 02-1 ST must be finished successfully, exit criteria are fulfilled
- 02-2 SIT entry criteria must be fulfilled, e.g.
  - Preparation of connected test systems or simulators
  - Test Scenarios are ready
  - Test Data are available
  - Special Test environment is ready

# Test Execution Test Stages - 02-3 NFR-Test





# Test Execution Test Stages - 02-3 NFR-Test



- What to test
  - Performance
  - Load
  - Operational
  - Security
- Typically only NFR Test experts are involved

# Test Execution Test Stages - 02-3 NFR-Test



- Hints how to proceed
  - For performance test a stable system is necessary but this is usually only obtainable at the end of testing
  - Recommendation: Random testing, focusing on Test Cases that will be executed often in the real system
  - Examine benefits versus costs (effort for scripting)





## Test Execution Test Stages - 03-1 UAT

- Entry criteria must be fulfilled
  - e. g. special environment
- Invitation to special testers, e. g. experts from the customer, final user
- Typically in this status the software is stable, so (hopefully) all the critical defects are already detected

## Test Execution 03-2 – Software acceptance



- Finally:
  - Based on the report by the Test Team decision by the responsible people
  - Software Acceptance

