Software Testing at a Glance – or two

The purpose of test:

Provide information to assure the quality of
- the product (finding faults)
- decisions (quantifying risks)
- process (finding root causes)

Mistake / Error

Defect / Fault

Failure

Test is comparing what is to what should be

WWW.DELTA.DK

Fault correction

Process improvement

Incident report

DELTA

Start here

Business goals

Business requirements

Vision / ideas / needs / expectations

Deployment

Maintenance testing

Tool support for monitoring Continuous analysis of usage Hyperlinks

Time

Disposal

Organizational management

Process Improvement

Project management

Test management

Configuration management

Quality assurance

Release!

Customer satisfaction

Test policy
- definition of test
- test process to use
- test evaluation
- quality level
- approach to improvement

Overall test strategy R

Test strategy
1. Entry & exit criteria
2. Test approach
3. Techniques to use
4. Completion criteria
5. Independence
6. Standards to use
7. Environment
8. Test automation
9. Degree of reuse
10. Retest and regression testing
11. Test process
12. Measures
13. Incident management

Development models

Sequential C D I T

Waterfall and V-models

Iterative

RAD, RUP or Agile

Conceive
Design
Implement
Test + Correction

Project plan R

Be prepared - plan ahead!

Requirements tower
- Environment requirements: Platform, Hardware, Standards
- Product quality requirements
- Functional requirements
- Non-functional requirements: Performance requirements, Usability requirements, Availability requirements, Maintainability requirements
- Project requirements: Cost, Resources, Time

Tool support for Requirements management

Network, Data, Software, Hardware, Services

Good code / Good test / Known quality

Poor code / Poor test / UN-known quality

Communication is

this is what I mean / this is what I think / this is what I say / this is what I hear / difficult / background / background

Phase specific test strategy R

Overall test plan R

Test plan
1. Introduction
2. Test item(s)
3. Features to be tested
4. Features not to be
5. Approach
6. Item pass/fail criteria
7. Suspension criteria
8. Test deliverables
9. Testing tasks
10. Environmental needs
11. Responsibilities
12. Staffing and training
13. Schedule
14. Risks

Test phases
- Acceptance testing
  - user
  - operational (OAT)
  - contractual (FAT)
  - alpha & beta
- System integration testing
- System testing
  - functional
  - non-functional
- Component integration testing
- Component testing

A plan must be Specific Measurable Agreed Relevant Time specific

Tester skills

Test skills, Soft skills, IT skills, Domain knowledge

Test Roles
- Test leader (manager)
- Test analyst / designer
- Test executer
- Reviewer / inspector
- Test environment responsible
- (Test)tool responsible

The best (test)team is based on knowledge of

Belbin types:
- Cerebral: Plant, Monitor/Evaluator, Specialist
- Action oriented: Shaper, Implementer, Completer/Finisher
- People oriented: Coordinator, Team-worker, Ressource investigator

Responsibility matrix

Tasks × People — R Responsible, P Performing, C Consulted

Test could be ruled by product risks + willingness to run risks

Triangle of test quality

Change one aspect -> change at least two

Resources, Time, Quality

Risk management

Risks hit in many places

Risk exposure = probability x effect

Business, Process, Project, Product

Development

User requirements → Product design → System requirements → Architectural design → Detailed design → Coding

Test plan R | **Test specification** R

Acceptance testing T → Report

System integration testing T → Report

System testing T → Report

Component integration testing T → Report

Component testing T D

Integration

Hw, Component, S

The price of a fault correction depends on when the fault is found

1000 / 100 / 10 / 1 — Requirement specification, Design, Test, Production — Source: Grove Consultants, UK

Tool support for dynamic analysis

Tool support for static testing
Static analysis, incl. requirements testing

Quality assurance for all work products

Definition of quality criteria → Validation → Verification → Reporting

R **Review types:** informal review, walkthrough, technical review, management review, inspection, audit

S **Static analysis** by tools D **Dynamic analysis** by tools T **Dynamic test** manually or by tools

Tool support — **Configuration management** of selected work products, product components, and products

Identification, Storage, Reporting, Recognition → Investigation → Action, Change management, Disposition

Tool support

Plan at project start

Plan during development

Plan at test start

Tool support for
Test planning and monitoring

Management (of everything)

Risk analysis → Estimation → Scheduling → Monitoring and control

Communication

correction of the test object

Tracing

Req. 146, Req. 147, Req. 148 → Test case 27, Test case 56

- prevents omissions
- prevents gold-plating
- shows the rationale for test cases

Incident report

Recognition: supporting data, classification, impact
Investigation: supporting data, classification, impact
Action: supporting data, classification, impact
Disposition: supporting data, classific., impact

Test process for all test phases

Test planning → Test specification → Test execution → Test recording → Checking for test completion

Monitoring and control

From test basis to test cases

Base document → derive → Test conditions / Test requirements → create using test case design techniques → Test cases → Organize into → Test procedure / Test script

Test logs

Test coverage

% of the coverage items touched by a test

Coverage items:
- statements, decisions, conditions
- interfaces
- user requirements, use cases
- business processes, business goals

Phase, Phase, Phase, Phase, Phase test report

Test case design techniques

Specification based
- equivalence class partitioning
- boundary value analysis
- decision tables
- state transitions
- use cases

Experience based
- error guessing
- check lists
- exploratory

Structure based
- statements
- decisions
- conditions
- models

Tool support for test specification

Test design
Test input data preparation
Test oracles

A fool with a tool is still a fool

Test Specification
1. Test Design
   - 1.1 Features to be tested
   - 1.2 Approach refinement
   - 1.3 Associated test case
   - 1.4 Feature pass/fail criteria
2. Test Cases
   - 2.1 Test Case 1
     - 2.1.1 Test items (rationale)
     - 2.1.2 Prerequisites
     - 2.1.3 Inputs
     - 2.1.4 Expected outcomes
     - 2.1.5 Inter case dependencies
   - 2.N Test Case n

Trace tables

Test Procedure
1. Purpose
2. Prerequisites
3. Procedure steps

The white-box insect

branch (in), basic block, statement, decision, branch point, condition, branch (out), branch (out), decision outcome

Overall test report

Test environment

Hardware, software, network, other systems, tools, data, room, peripherals, stationary, chairs, 'phones, food, drink, candy

Tool support for test execution and logging

Test running, Test harness and drivers, Comparators, Simulator, Test script generation, Coverage measurement, Security testing

Tool support for non-functional test Performance, Load, Stress

Test report
1. Summary
2. Variances
3. Comprehensiveness
4. Summary of results
5. Evaluation
6. Summary of activities

Test targets

New product
- functional
- non-functional
- structure/architecture

Maintenance

Change related
- regression testing
- retesting

Why do we work?

incentives, motives

General process improvement models

CMM - levels: 1 Initial, 2 Repeatable, 3 Described, 4 Controlled, 5 Optimising — Higher productivity and quality, Lower risk

CMMI –process areas

ISO 15504 – result profile

Process improvement

Learning → Acting (implementing) → Establishing (plan and write) → Diagnosing (assessment) → Initiating → Learning

Test process improvement models

TMM - levels

TPI® - matrix — Scale, Key areas

Cooperation matrix

Maturity levels 1 2 3 4 5

procedures methods techniques, Learning, Tools, Actions, Adjustment, Deployment

Input Entry criteria — **Process** template — Procedure: Activity 1, Activity 2, - Activity n — **Exit** criteria — **Output**

Test approaches

Analytical (risk), Model-based (statistical), Structured
Consultative (domain experts), Methodical (check-lists), Heuristic (exploratory)
Standard compliant, Regression-averse (reuse and automation)

Degree of independence (producer vs. tester)
1. Producer tests own product
2. Tests are designed by other team members
3. Tests are designed by independent testers
4. Tests are designed by external testers

Standards

Quality assurance (you shall test)
– ISO 9000:2001

Industry specific (you shall test this much)
– aviation
– medical devices
– military equipment

Testing standards (you shall test like this)
– BS-7925
– IEEE 829
– IEEE 1028

Other related standards
– IEEE 1044
– ISO 9126
– IEEE/ISO 12207
– IEEE/ISO 15288

If you don't measure you're left with only one reason to believe you're in control – hysterical optimism.
Tom De Marco

The golden rule of testing:
Always test so that whenever you have to stop you have done the best possible test.

Graphic advise: EBB-CONSULT, Denmark

Screen Beans® Clip Art, USA

V.T-UK-o-0803