MTAT.03.159: Software Testing

Lecture 07:
Test Documentation, Organisation and Process Improvement
Exam Preparation

Spring 2017

Dietmar Pfahl
email: dietmar.pfahl@ut.ee
Structure of Lecture 07

- Test Planning & Documentation
- Test Organisation
- Test Process Improvement (TMMi)
- Exam Preparation
Test Planning

• Objectives
• What to test
• Who will test
• When to test
• How to test
• When to stop

Elective course (Fall’17): Hands-on SW Testing MTAT.03.294
IEEE 829-2008: Standard for Software and System Test Documentation

FIG. 7.4
Software Project Management Plan

Goals
- Business
- Technical
- Business/technical
- Political

Quantitative/qualitative

Policies
- High-level statements of principle or courses of action
- Govern the activities implemented to achieve stated goals
Hierarchy of Test Plans

### Test Plan Components

1. Test plan identifier
2. Introduction
3. Items to be tested
4. Features to be tested
5. Approach
6. Pass/fail criteria
7. Suspension and resumption criteria
8. Test deliverables
9. Testing tasks
10. Test environment
11. Responsibilities
12. Staffing and training needs
13. Scheduling
14. Risks and contingencies
15. Testing costs
16. Approvals

---

- **Software quality assurance (V&V) plan**
- **Master test plan**
  - **Unit test plan**
  - **Integration test plan**
  - **System test plan**
  - **Acceptance test plan**
- **Review plan: Inspections and walkthroughs**
Test plan according to IEEE Std 829-2008 (Appendix II)

a) Test plan identifier  
b) Introduction  
c) Test items  
d) Features to be tested  
e) Features not to be tested  
f) Approach  
g) Item pass/fail criteria  
h) Suspension criteria and resumption requirements  
i) Test deliverables  
j) Testing tasks  
k) Environmental needs  
l) Responsibilities  
m) Staffing and training needs  
n) Schedule  
o) Risks and contingencies  
p) Approvals
Test Plan (1)

a) Test plan identifier
b) Introduction
   – Product to be tested, objectives, scope of the test plan
   – Software items and features to be tested
   – References to project authorization, project plan, QA plan, CM plan, relevant policies & standards
c) Test items
   – Test items including version/revision level
   – Items include end-user documentation
   – Defect fixes
   – How transmitted to testing
   – References to software documentation
Test Plan (2)

d) Features to be tested
   – Identify test design / specification techniques
   – Reference requirements or other specs

e) Features not to be tested
   – Deferred features, environment combinations, …
   – Reasons for exclusion

f) Approach
   – How you are going to test this system
     • Activities, techniques and tools
   – Detailed enough to estimate
   – Completion criteria (e.g. coverage, reliability)
   – Identify constraints (environment, staff, deadlines)
Test Plan (3)

g) Item pass/fail criteria
- What constitutes success of the testing
- Coverage, failure count, failure rate, number of executed tests, …
- Is NOT product release criteria

h) Suspension and resumption criteria
- For all or parts of testing activities
- Which activities must be repeated on resumption

i) Test deliverables
- Test plan
- Test design specification, Test case specification
- Test procedure specification, Test item transmittal report
- Test logs, Test incident reports, Test summary reports
j) Testing tasks
   - Including inter-task dependencies & special skills
   - Estimates
k) Environment
   - Physical, hardware, software, tools
   - Mode of usage, security, office space
   - Test environment set-up
l) Responsibilities
   - To manage, design, prepare, execute, witness, check, resolve issues, providing environment, providing the software to test
m) Staffing and Training needs
Test Plan (5)

n) Schedule
   – Test milestones in project schedule
   – Item transmittal milestones
   – Additional test milestones (environment ready)
   – What resources are needed when

o) Risks and Contingencies
   – Testing project risks
   – Contingency and mitigation plan for each identified risk

p) Approvals
   – Names and when approved
Test Case Specification – Why?

• Organization
  – All testers and other project team members can review and use test cases effectively

• Repeatability
  – Know what test cases were last run and how so that you could repeat the same tests

• Tracking
  – What requirements or features are tested?
  – Tracking information’s value depends on the quality of the test cases

• Evidence of testing
  – Confidence (quality)
  – Detect failures
## Defect Report

*(Test incidence report)*

- **Summary**
- **Incident Description**
- **Impact**

### Lab 1

<table>
<thead>
<tr>
<th>Bug 1466</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Summary:</strong> Quick Quote &amp;秸秆 Postcode works only once after page load</td>
</tr>
<tr>
<td><strong>Product:</strong> [Product]</td>
</tr>
<tr>
<td><strong>Components:</strong> [Component]</td>
</tr>
<tr>
<td><strong>Status:</strong> [Status]</td>
</tr>
<tr>
<td><strong>Severity:</strong> [Severity]</td>
</tr>
<tr>
<td><strong>Environmental:</strong> Lab 1</td>
</tr>
</tbody>
</table>

#### Additional Info

- **Bug advocacy:** Naturally, manually refreshing the page to get the address retrieval working again clears all data from the fields except for same and state if blank, for some reason.

- **Bug advice:**
  - [Comment 1]
  - [Comment 2]

---

MTAT.03.159 / Lecture 07 / © Dietmar Pfahl 2017
Defect Report (Test incidence report)

Summary

• This is a summation/description of the actual incident.
  – Provides enough details to enable others to understand how the incident was discovered and any relevant supporting information

• References to:
  – Test Procedure used to discover the incident
  – Test Case Specifications that will provide the information to repeat the incident
  – Test logs showing the actual execution of the test cases and procedures
  – Any other supporting materials, trace logs, memory dumps/maps etc.
Defect Report (Test incidence report)

Incident Description

• Provides as much details on the incident as possible.
  – Especially if there are no other references to describe the incident.
• Includes all relevant information that has not already been included in the incident summary information or any additional supporting information

• Information:
  – Inputs
  – Expected Results
  – Actual Results
  – Anomalies
  – Date and Time
  – Procedure Step
  – Attempts to Repeat
  – Testers
  – Observers
Defect Report  
(Test incidence report)

Impact

- Describe the actual/potential damage caused by the incident.
  - Severity
  - Priority

- Severity and Priority need to be defined so as to ensure consistent use and interpretation, for example:

  - Severity – The potential impact to the system
    - Mission Critical - Application will not function or system fails
    - Major - Severe problems but possible to work around
    - Minor – Does not impact the functionality or usability of the process but is not according to requirements/design specifications

  - Priority – The order in which the incidents are to be addressed
    - Immediate – Must be fixed as soon as possible
    - Delayed – System is usable but incident must be fixed prior to next level of test or shipment
    - Deferred – Defect can be left in if necessary due to time or costs
Test results report

- Test cases executed
- Versions tested
- Defects found and reported

![Problem Status by Resolution](chart)

Figure 6. Defect distributions, showing number of defects detected over time, for the three studied projects. Ship dates indicated.
Standards

• IEEE 829-2008
  Standard for Software Test Documentation
• IEEE 1008-1993
  Standard for Software Unit Testing
• IEEE 1012-2012
  Standard for System and Software Verification and Validation

->
• ISO/IEC/IEEE 29119 Software Testing (5 parts)
  – replaces most of the older standards
Structure of Lecture 07

• Test Planning & Documentation
• Test Organisation
• Test Process Improvement (TMMi)
• Exam Preparation
7 approaches to test organisation

1. Each person’s responsibility
2. Each unit’s responsibility
3. Dedicated resource
4. Test organisation in QA
5. Test organisation in development
6. Centralized test organisation
7. Test technology centre

[Kit, Software Testing in the Real World Ch 13, 1995]
7 approaches to test organisation

1. Each person’s responsibility
2. Each unit’s responsibility
3. Dedicated resource
4. Test organisation in QA
5. Test organisation in development
6. Centralized test organisation
7. Test technology centre

[Kit, Software Testing in the Real World Ch 13, 1995]
Which organization should we choose?

- Depending on
  - size
  - maturity
  - focus

- The solution is often a mix of different approaches
Watch James Bach’s open lecture video (course wiki)!
Structure of Lecture 07

• Test Planning & Documentation
• Test Organisation
• Test Process Improvement (TMMi)
• Exam Preparation
Process quality and product quality

- Quality in process
  Quality in product

- Project:
  - instantiated process

- ISO 25000:
  - *Process quality* contributes to improving *product quality*, which in turn contributes to improving *quality in use*
Process improvement models vs Test Process improvement models

• (Integrated) Capability maturity model (CMM, CMMI)
• Software process improvement and capability determination (SPICE)
• ISO 9001, Bootstrap, …

Test Process Improvement Models:
• Test maturity model (TMM, TMMi)
• Test process improvement model (TPI)
• Test improvement model (TIM)
• Minimal Test Practice Framework (MTPF)
• …
Test Maturity Model (TMM)

- Levels
- Maturity goals and sub-goals
  - Scope, boundaries, accomplishments
  - Activities, tasks, responsibilities
- Assessment model
  - Maturity goals
  - Assessment guidelines
  - Assessment procedure
Level 2: Phase Definition

- Institutionalize basic testing techniques and methods
- Initiate a test planning process
- Develop testing & debugging tools
Level 3: Integration

- Control and monitor the testing process
- Integrate testing into software lifecycle
- Establish a technical training program
- Establish a software test organization
Level 4: Management and Measurement

- Software quality evaluation
- Establish a test management program
- Establish an organization-wide review program
Level 5: Optimizing, Defect Prevention, and Quality Control

- Test process optimization
- Quality control
- Application of process data for defect prevention
Recommended Textbook Exercises

• Chapter 14
  – 2, 4, 5, 6, 9
• Chapter 9
  – 2, 3, 4, 5, 8, 12
• Chapter 7
  – 2, 3, 6, 8, 9, 11
• Chapter 8
  – 2, 3, 6, 7, 9
• Chapter 16
  – No exercises
Structure of Lecture 07

• Test Planning & Documentation
• Test Organisation
• Test Process Improvement (TMMi)
• Exam Preparation
Exam Dates

- Exam 1: Tue 09-May, 15:15-17:15, room 405 – max. 46 students
- Exam 2: Fri 12-May, 14:15-16:15, room 404 – max. 35 students
- Exam 3: Mon 15-May, 15:15-17:15, room 405 – max 46 students

You must receive

... at least 20 marks from the homework assignments (labs 1-6) to qualify for the exam and
... at least 10 marks in the exam to not fail the course.
In total, you need at least 50 marks to not fail the course.

- Retake Exam (resit): 23-May, 14:15-16:15 (J. Liivi 2-612)
  - Please note that you must register for the retake exam at the latest 3 days before the exam date
Final Exam

Admission Rule:
• At least 20 marks from labs 1-6

Minimum required to not fail the course:
• At least 20 marks from labs AND
• At least 10 marks in exam AND
• At least 50 marks overall
Questions ?
Final Exam – Format

• Written exam (40% = 40 marks)
  – Based on textbook, lectures and lab sessions
  – Open book
  – 100 min
  – 2 Parts:
    • Part 1 – Multiple-Choice (10 marks)
    • Part 2 – Constructive Tasks (30 marks)
      – Answers might require some technical work & calculation

Example exams from 2013 to 2016 are posted on the course wiki under tab ’Assessment and Grades’
Questions?
Final Exam – Content/Topics Overview

• Introduction to Software Testing
• Black-Box Testing Techniques
• White-Box Testing Techniques
• Mutation Testing
• Static Testing (InspectionReview) and Defect Estimation
• Test Tools (and Measures)
• Test Documentation, Organisation, Improvement
Final Exam – Content/Topics Overview

Introduction to Software Testing:
• Know the basic terminology
  – Software Testing & Software Quality
  – Verification & Validation
  – Error – Fault – Failure
    • NB: Two competing definitions of ’Error’
  – Test Case – Test Suite – Test Oracle ...
  – Test Levels
  – Issue Reporting
Final Exam – Content/Topics Overview

Black-Box Testing Techniques:
• Difference between Black-Box and White-Box Testing
  – Strengths & Weaknesses of each
• Know various BBT Techniques (Methods):
  – Equivalence Class Partitioning
  – Boundary Value Testing
  – Combinatorial Testing
  – State-Transition-Testing
  – Exploratory Testing
  – Usability Testing
Final Exam – Content/Topics Overview

White-Box Testing Techniques:

• Difference between Black-Box and White-Box Testing
  – Strengths & Weaknesses of each
• Control-Flow Testing
  – Know how to construct a Control-Flow-Graph
  – Know different coverage criteria:
    • Statement/Block, Decision/Branch, Condition, Linearly Independent Paths, etc.
• Data-Flow Testing
• Mutation Testing
Final Exam – Content/Topics Overview

Static Testing (Inspection) and Defect Estimation:

• Document Reviews (Inspections)
  – Why needed?
  – What variants exist?

• Static Code Analysis
  – What are false positives?

• Defect Estimation:
  – Capture-Recapture Model
Final Exam – Content/Topics Overview

Test Lifecycle:
• Agile Testing
• Specifics of Testing OO Code
  – Intra-Class Testing (‘Stack’ Example)
  – Inter-Class Testing
  – State-Transition-Testing
• System versus Unit Testing
• Regression Testing
Final Exam – Content/Topics Overview

Tools (and Measures):
• Different Tools for Different Purposes!
• Test Automation
  – Capture-Replay (Web-Testing / GUI Testing)
  – Automated Regression Testing
• Test Measures
  – Test Coverage, Test Effectiveness, Test Efficiency, etc.
Questions ?
Thank You!