Course Outline (as of 09-Apr-2014)

<table>
<thead>
<tr>
<th>Numerical course code</th>
<th>MTAT.03.159</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Software Testing</td>
</tr>
<tr>
<td>Faculty/Department</td>
<td>Faculty of Mathematics and Computer Science, Institute of Computer Science, Chair of Software Systems</td>
</tr>
<tr>
<td>Amount of credits (1 ECTS = 26 hours)</td>
<td>3 ECTS</td>
</tr>
<tr>
<td>Amount of credits (before 31-Aug-2009)</td>
<td>2 CP</td>
</tr>
<tr>
<td>CP*1.5 (before 31-Aug-2009)</td>
<td>3</td>
</tr>
<tr>
<td>Duration in semesters</td>
<td>1</td>
</tr>
<tr>
<td>Final assessment</td>
<td>Differentiated</td>
</tr>
<tr>
<td>Course responsible</td>
<td>Dietmar Pfahl</td>
</tr>
<tr>
<td>Course instructors</td>
<td>Dietmar Pfahl, Briti Deb, Svetlana Omelkova</td>
</tr>
<tr>
<td>Course language</td>
<td>English</td>
</tr>
<tr>
<td>Study levels</td>
<td>Bachelor's studies</td>
</tr>
<tr>
<td>Forms of teaching and learning</td>
<td>Lectures (incl. practical work / labs): 32 hours Independent work (incl. e-learning): 46 hours</td>
</tr>
<tr>
<td>Study period</td>
<td>Weeks 32-39 First lecture: 10-April-2013 Last lecture: 22-May-2013 Labs (5 groups) start in week 33</td>
</tr>
</tbody>
</table>

Prerequisites
Compulsory: MTAT.03.094 Software Engineering (6 ECTS, 4 CP)
Recommended: MTAT.03.130 Object-oriented Programming (6 ECTS, 4 CP)

Curricula containing this course
Computer Science (2476)    bac. 2011/12 2012/13 2013/14 2014/15
Information Technology (2477)  bac. 2011/12
Software Engineering (100864)  mas. 2011/12 2012/13 2013/14

Objectives
The course addresses the essential concepts of software quality control and testing and introduces various testing strategies and types of testing. It will also give an overview of different software defects, software defect management, and organizational aspects of software testing.

Learning outcomes
On successful completion of this course, students will be able to demonstrate knowledge of:
- The role of testing in the software development process
- Test planning and documentation
- Different types of testing techniques
- Static testing and defect estimation

Course elements
The course will cover:
1. Black-box testing and white-box testing techniques
2. Combinatorial testing
3. Usability testing
4. Certification testing
5. Exploratory testing  
6. Reliability growth models  
7. Inspections and reviews  
8. Static code analysis  
9. Test measures  
10. Test Maturity Model (TMM)  
11. Test organization  
12. Test tools

**Schedule (tentative – adjustments will be announced on the course wiki)**

- Week 32: Introduction to Software Testing  
- Week 33: Black-box test techniques  
- Week 34: White-box test techniques  
- Week 35: no lecture (holiday on May 1)  
- Week 36: Static Testing (Inspection) and Defect Estimation  
- Week 37: Lifecycle, Documentation, Organisation, Tools  
- Week 38: Metrics and Test Process Improvement (Test Maturity Model)

**Assessment**

- Labs and associated assignments/reports (homework) – work in pairs (40% of course grade)  
- Project (literature study) – work in groups of 3 (20% of course grade)  
- Written exam – individual (40% of course grade)