# Course Outline (as of 27-August-2015)

<table>
<thead>
<tr>
<th>Numerical course code</th>
<th>MTAT.03.094</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Software Engineering</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>6 ECTS</td>
</tr>
<tr>
<td>Amount of credits (before 31-Aug-2009)</td>
<td>4 CP</td>
</tr>
<tr>
<td>CP*1.5 (before 31-Aug-2009)</td>
<td>6</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, Yar Muhammad, Svetlana Omelkova, Kristiina Rahkema, Sander Soo</td>
</tr>
<tr>
<td>Course language</td>
<td>English (lecture/exam) and English/Estonian (lab/practice sessions)</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): 56 hours Independent work: 100 hours</td>
</tr>
<tr>
<td>Study period</td>
<td>16 Weeks</td>
</tr>
<tr>
<td></td>
<td>First lecture: 04-Sep-2015 (week 1)</td>
</tr>
<tr>
<td></td>
<td>Last lecture: 11-Dec-2015 (week 15, tentative)</td>
</tr>
<tr>
<td></td>
<td>First lab: 07-Sep-2015 (week 2)</td>
</tr>
<tr>
<td></td>
<td>Last lab: 16-Dec-2015 (week 16, tentative)</td>
</tr>
</tbody>
</table>

### Prerequisites
Compulsory: MTAT.03.130 Object-oriented Programming (6 ECTS, 4 CP)

### Curricula containing this course
- Computer Engineering (83376) mas. 2012/2013 2013/2014
- Mathematical Statistics (2474) bac. 2015/2016

### Objectives
To obtain basic knowledge in software engineering and primary skills for working at any stage of software development projects.

### Learning outcomes
Upon successful completion of this course, students will be able to demonstrate basic knowledge of and skills in:
- software engineering paradigms;
- system analysis;
- requirements analysis;
- planning;
- implementation;
- quality assurance (verification and validation; testing);
- maintenance (evolution);
- project management;
- software processes and methodology.
Course elements
The course covers:
- software engineering paradigms
- project management
- system and requirements analysis
- design
- implementation
- testing
- quality management and control

Schedule of lectures (tentative – adjustments will be announced on the course wiki)
- Week 01: Lecture 01 - Introduction to Software Engineering
- Week 02: Lecture 02 - Requirements Engineering - I
- Week 03: Lecture 03 - Requirements Engineering - II
- Week 04: Lecture 04 - Analysis
- Week 05: Lecture 05 - Development Infrastructure - I
- Week 06: Lecture 06 - Development Infrastructure - II
- Week 07: Lecture 07 - Architecture and Design
- Week 08: -- (no lecture)
- Week 09: Lecture 08 - Refactoring
- Week 10: Lecture 09 - Verification and Validation - I
- Week 11: Lecture 10 - Verification and Validation - II
- Week 12: Lecture 11 - Agile/Lean Methods
- Week 13: Lecture 12 - Software Quality Management
- Week 14: -- (no lecture)
- Week 15: Lecture 13 - Measurement / Course wrap-up, review and exam preparation
- Week 16: -- (no lecture)

Schedule of labs (tentative – adjustments will be announced on the course wiki)
- Week 01: -- (no labs)
- Weeks 02-04: Requirements gathering
- Weeks 04-06: Formalizing, modeling, planning
- Weeks 06-08: Development infrastructure
- Weeks 08-10: Realization - I
- Weeks 10-12: Realization - II
- Weeks 12-14: Automatic testing and refactoring
- Weeks 14-16: Verification and validation

Assessment
- Practice work / Labs (70% of total course grade) – group work (target group size: 3 students per group)
- Written exam (30% of course grade) – individual / open book

Recommended Literature
Additional literature will be announced on the course wiki.