Master's Thesis (120 credits) in Software Engineering

30 ECTS credit points (30 högskolepoäng)

Course code: PA2534
Educational level: Advanced level
Course level: A2E
Field of education: Technology
Subject group: Computer Technology

1 Course title and credit points
The course is titled Master's Thesis (120 credits) in Software Engineering/Masterarbete i Programvaruteknik and awards 30 ECTS credits. One credit point (högskolepoäng) corresponds to one credit point in the European Credit Transfer System (ECTS).

2 Decision and approval
This course is established by Head of Department of Software Engineering 2014-06-30. The course syllabus was revised by Head of Department of Software Engineering and applies from 2016-08-29. Reg. no: BTH-2.4.2-0235-2016.

3 Objectives
The aim of the course is that students will practice their ability to define, plan, implement and present independent scientific research work. Students are expected to apply the knowledge acquired during their education and independently immerse themselves in one or more specific sub-areas within the main subject area software engineering. The goal of the student is to, orally and in writing, report the results of his or her research and to critically examine and oppose on another thesis. The presentations shall comply with the requirements and criteria for academic writing.

4 Content
The course comprises four components:
1. Prestudy and planning
   a. Research work
   b. Supervision and progress tracking
   c. Written reporting
2. Implementation
3. Oral presentation and defense
4. Opposition
   a. Oral opposition
   b. Written opposition

Prestudy and planning include developing a project plan that includes a timetable for the project. In the research phase (implementation) the project is executed and documented in an academic report according to the timetable. This also includes regular progress tracking of the work. At the oral presentation, the student will defend his or her thesis. Each student must also act as opponent on another thesis.

5 Aims and learning outcomes
Knowledge and Understanding
After the course the student should be able to:
• demonstrate a broad knowledge and understanding in software engineering,
• show substantially deeper knowledge of at least one sub-area of software engineering, including a deeper insight into current research and development,
• demonstrate deeper methodological knowledge and understanding within software engineering.

Skills and Abilities
After the course the student should be able to:
• demonstrate the ability to creatively, critically and independently identify and formulate scientific questions,
• demonstrate the ability to answer scientific questions using suitable methods and thereby contribute to the development of knowledge in software engineering,
• demonstrate the ability to plan, monitor and complete an independent research project within a given timeframe and supervision resources,
• discuss the thesis orally and in writing clearly explain and discuss his or her conclusions and the knowledge and arguments behind them, in dialogue with researchers, students at the same level in the field and lay people,
• discuss the thesis orally and in writing in English,
• demonstrate the ability to systematically and on a scientific basis apply knowledge acquired during his
or her education and to identify, analyze, synthesize and critically examine scientific literature.

**Values and attitudes**

After the course the student should be able to:
- demonstrate the ability to systematically and critically examine both their own and others' work in relation to relevant scientific, technical, social, and ethical aspects;
- demonstrate the ability to identify their need of further knowledge and to take responsibility for their learning.

**6 Learning and teaching**

The student works individually and independently, and is responsible for completing a thesis with sufficiently high quality within the given timeframe and with given supervision resources. Exceptions to individual work must be approved by the examiner. Each student is supervised by an academic supervisor at the university. In addition to the academic supervisor, a student might have an external supervisor from industry or another university.

The course starts with introductory seminars. The remaining time consists of independent work in the form of planning, executing, monitoring, and reporting a thesis and an opposition against another thesis. During the work, the student should maintain an e-portfolio in the course’s learning platform where he or she regularly informs about the status and progress of his or her degree project. The final, revised academic report is graded by the examiner after oral presentation and defense. The examiner grades the academic report based on his or her own assessment and consideration of independent peer reviews. Also the project plan is graded by the examiner supported by independent peer reviews.

The presentation and defense can only be carried out when (i) there is an approved project plan and (ii) the academic report is sufficient in its current form for presentation and defense. This assessment does not mean that the report will be approved, since grading is done by the examiner after presentation and defense of the academic report. The project plan, oral presentation and defense, oral and written opposition, and the academic report shall follow the instructions and templates included in the Instructions for Degree Projects provided by the Faculty of Computing.

The course is a campus course. The student is expected to be available for supervision. It is the student’s responsibility to conserve supervision resources. The student is not entitled to supervision time outside term time. A student who does not complete his or her thesis within the term it was commenced, can get continued supervision for a limited time only and no more than twelve months after the course ends. The examiner has the right to end supervision when all supervision time is exhausted. However, a student is always entitled to submit his or her independent work for grading at subsequent examination times. A students who re-registers for the course is not entitled to renewed supervision time.

The teaching language is English.

**7 Assessment and grading**

**Examination of the course**

<table>
<thead>
<tr>
<th>Code Module</th>
<th>Credit</th>
<th>Grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>1410 Project plan</td>
<td>2 ECTS</td>
<td>G-U</td>
</tr>
<tr>
<td>1420 Presentation/Defense</td>
<td>1 ECTS</td>
<td>G-U</td>
</tr>
<tr>
<td>1430 Thesis opposition</td>
<td>1 ECTS</td>
<td>G-U</td>
</tr>
<tr>
<td>1440 Thesis and Realisation</td>
<td>16 ECTS</td>
<td>A-F</td>
</tr>
</tbody>
</table>

1 Determines the final grade for the course, which will only be issued when all components have been approved.

The course will be graded A Excellent, B Very good, C Good, D Satisfactory, E Sufficient, FX Insufficient, supplementation required, F Fail. Grading is performed by the examiner after considering independent peer reviews.

A student, who at the end of the course has not submitted a project plan is given the grade U on component Project plan (1410), since the student did not demonstrate adequate ability to perform tasks within specified timeframes.

A student, who has not submitted an academic report of sufficient quality to be presented and defended within 12 (16) months from the start of the course, can at most receive grade B (C), since the student did not demonstrate adequate ability to perform tasks within specified timeframes.

The oral presentation and defense of the thesis and oral opposition shall take place on campus.

The number of times that a student may be examined to get a passing grade on each of the examination components of the course is limited to five.

**8 Course evaluation**

The course coordinator is responsible for systematically gathering feedback from the students in course evaluations and making sure that the results of these feed back into the development of the course.

**9 Prerequisites**

Admission to the course requires a minimum of 90 higher education credits in Software Engineering, including at least 30 credits at the advanced level. In addition, students must have successfully completed a course in Research Methods in Software Engineering and/or Computer Science for 7.5 credits.

**10 Field of education and subject area**

The course is part of the field of education and is included in the subject area Software Engineering.

**11 Restrictions regarding degree**

The course cannot form part of a degree with another course, the content of which completely or partly corresponds with the contents of this course.

**12 Additional information**
Replaces PA2403, PA2503 and PA2511.

13 Course literature and other teaching material

Master’s Thesis (120 credits) in Software Engineering

Textbooks
Instructions for Degree Projects (provided by university department).
Other textbooks are chosen individually by the student in consultation with the supervisor.

Reference literature

Authors: Mikael Berndtsson et al.
Publisher: Springer
Year: 2007, Pages: 162

2. Experimentation in Software Engineering – An Introduction; 2nd Edition
Authors: C. Wohlin, P. Runeson, M. Höst, M.C. Ohlsson, B. Regnell, A. Wesslén
Publisher: Springer Verlag
Year: 2012, Pages: 250