[Mallversion 9.2, fastställd 2014-10-29 av vicerektor och dekanerna gemensamt. För nya och reviderade kurser från och med ht2014.]

Blekinge Institute of Technology
Fakulty of Computing

COURSE SYLLABUS

Masterarbete i programvaruteknik

Master's Thesis (120 credits) in Software Engineering

30 higher education credits (30 ECTS credit points)

Course code: PA2534
Level: Advanced level
Depth: A2E
Education area: Technology
Area group: Computing

Main area: Software Engineering
Version: 3
Valid from: 2014-09-01
Decided on: 2014-06-30 (Swedish version)
Replaces syllabus from: 2013-12-11

1. Course name and credits

The course is titled Master Thesis in Software Engineering / Master's Thesis (120 credits) in Software Engineering and comprises 30 higher education credits. A credit corresponds to a credit in the European Credit Transfer System (ECTS).

2. Decision

This course is established by the head of the Department of Software Engineering 2014-06-30. The syllabus has been revised by the head of the Department of Software Engineering and is valid from 2014-09-01.
Ref: BTH 4.1.1-0390-2014
Replaces PA2511

3. Aim

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 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. Contents

The course comprises four components:

1. Prestudy and planning (Project Plan)
2. Execution (Academic Report)
   a. Research and development
   b. Supervision
   c. Written presentation
3. Oral presentation and defense
   a. Opposition
   b. Written opposition
4. Oral opposition

5. Goals

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 an ability to answer scientific questions using suitable methods within a given time frame and supervision resources,
- demonstrate the ability to independently carry out research work and thereby contribute to the development of knowledge in software engineering,
- demonstrate the ability to present and discuss their conclusions as well as the knowledge and arguments that form the basis for those and to do so clearly, orally and in writing, in dialogue with

[Se **Riktlinjer för kursplaner** för instruktioner, information och bakgrund till de olika delarna i kursplanen.]

researchers as well as with students at the same level in the field and lay people,

- discuss the thesis orally and in writing in English,
- demonstrate the ability to apply the knowledge acquired during their education and to identify, analyze, synthesize and critically evaluate scientific literature in a scientifically-based and systematic way.

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, 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. Even 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 supervisor has notified the examiner in writing that the report is sufficient in its current form for presentation and defense.

The supervisor notification 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.

Teaching is in English.

7. Assessment and grading

Examination components of the course

Code	Title	Amount	Grade
1440	Report and implementation[1]	26 ECTS	A-F
1420	Presentation/Defense	1 ECTS	G-U
1410	Project plan	2 ECTS	G-U
1430	Opposition	1 ECTS	G-U

[1] Determines the final grade, which is issued only when all components have been approved.

The course will be graded using grades A Excellent, B Very good, C Good, D Satisfactory, E Acceptable, FX Fail–complementation required, F Fail.

Grading is performed by the examiner after considering independent peer reviews.

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 responsible is responsible for gathering students' comments on the course systematically and regularly and that the results of the evaluations in various forms affect the course's design and development.

9. Course 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. Education area and main area
The course belongs to education area Technology and is part of the main area Software Engineering.

11. Degree restrictions
The course cannot be included in a degree together with other course whose contents completely or partly overlap with the contents of this course.

12. Other
Replaces PA2403, PA2503 and PA2511.

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

1. Thesis Projects: A Guide for Students in Computer Science and Information Systems; 2nd Edition
   Authors: Mikael Berndtsson et al.
   Publisher: Springer
   Year: 2007, Pages: 162
   ISBN-13: 978-1848000087
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
   ISBN-13: 978-3642290435