robert.feldt@chalmers.se 2012-09-08

Scientific Publication in Software Engineering, 3 HEC

This PhD course gives an initial overview of scientific publication and trains students in the key processes involved in being a researcher in Software Engineering. The course is given as a series of seminars. Prior to each seminar, students work on an assignment specific to the seminar focus. Assignments should be close to the research topic/project/area of the students so that results are of a direct benefit.

Course credits:	3 ECTS credits (corresponding to ~80 hours of work in total)
Level:	PhD
Examiner/teacher:	Robert Feldt, robert.feldt@chalmers.se
Teaching methods:	Seminars with student preparations
Examination:	Completion of all assignments
Focus:	Train students in basic scientific publishing skills

1. General prerequisites

The requirement for the course is that you have been admitted to a PhD program in Software Engineering, and thus have completed a Master degree in Software Engineering, Computer Science or equivalent.

2. Course content

The course is organised around four themes that gradually give more specific advice on how to design, write, read and evaluate good Software Engineering papers:

1. Scientific Processes and Community,
2. Publication Processes and Strategies,
3. Reading and Reviewing Scientific Papers,
4. Research Methods

For all themes we cover briefly the general concept and definitions and then give concrete examples from the Software Engineering area and scientific community.

The first theme is focused on what science is, how it works in practice and the role of the scientific paper and community. We discuss the importance of publication and peer review and the difference between journal, conference and workshop publications at different levels. We also cover bibliometrics and scientometrics and discuss what is good and bad science and how it is evaluated and measured.

The second theme goes into more detail on publication processes and how to select and target the right scientific forum. Different rankings of publication fora are discussed. We also cover differences in the writing and publication process of a journal and conference paper.

The third theme is focused on how to read and evaluate scientific papers in Software Engineering. We discuss the elements of a scientific study, and thus a paper, and the major alternatives available for each. Checklists and advice on how to evaluate and review the quality of the different parts of a paper are covered. We also discuss reproducability and threats to validity.

In the fourth theme we go into some more detail on the main types of research approaches and methods used in Software Engineering. In particular we discuss quantitative and qualitative research, theoretic and empirical research, primary and secondary research, experiments, case studies, action research, systematic reviews and maps, interviews, and questionnaires. However, the information is cursory with the intent of being able to read, understand and comment on different types of methods; not on being able to design research using all of the methods.

3. Learning outcomes

After completion of the course the students is expected to be able to:

1. Read SE research papers and describe their motivation, methods used and the results
2. State the strong and weak points of a SE study, the threats to validity as well as how the study can be improved
3. Identify the research paradigm and method used in a SE study and compare it to alternative methods
4. Write good and constructive reviews of scientific papers in SE
5. Apply review templates objectively even on their own research studies and papers
6. Select relevant publication targets for their research
7. Describe the strongest publication targets for SE in general as well as for their sub-area
8. Write and update publication strategies for their research
9. Define bibliometrics used to rank publication targets, journals and researchers

4. Required reading

The course reading is based on papers for particular sub-topics. These theme specific research papers are assigned prior to each seminar. Separate reading lists are supplied for each course occasion; see the course home page. Example of material included in the course:

Alan Smith, “Task of the Referee”, IEEE Computer, Vol. 23, Issue 4, pp. 65-70, April 1990.
Mary Shaw, “Writing Good Software Engineering Research Papers”, pp. 726-736, 25th Int. Conf. on SE (ICSE), 2003.
Michael Ernst, “Reproducible tests? Non-duplicable results in testing and verification”, Int. Conf. on Software Testing (ICST), 2012.
Sense About Science, “Peer Review - The Nuts and Bolts”, www.senseaboutscience.org, 2012.

5. Assessment

The course is examined by the completion of the tasks that students do individually in preparation for each seminar as well as by the active participation of students on the seminars. Updates required to assignments after teacher comments needs to be resubmitted within two weeks after receiving the comments.

6. Grading scale

The course is graded with the following marks: Fail (F) or Pass (P).

7. Course evaluation

The course is evaluated in a discussion after the last seminar.

8. Additional information

The course is held in English and all assignments are written in English. The course points can be extended for students (in agreement with the examiner and supervisor) if more extensive assignment reports are submitted. This should be agreed upon at the start of the course.