Curriculum Vitae - Pedagogical Portfolio

1 Teaching

I have been the teacher in charge for the following courses

- Aalto University (formerly Helsinki University of Technology), Introduction to Software Engineering[1], 5 ECTS[2], (2011), 167 students, bachelors level
- Aalto University, Software Testing and Quality Assurance[3], 5 ECTS (2008-2010), 25 students, masters and postgraduate level
- Aalto University, Seminar in Software Engineering[4] , 5 ECTS (2003 and 2006), 10 students, masters and postgraduate level

I have also been teaching assistant in the following courses

- Aalto University, Software Development Project[5] 5-8 ECTS (2001-03, 2005-07), 150 students. This project course exits on both bachelor/undergraduate and master/postgraduate level
- Software Project Management[6] 5 ECTS (2002) 60 students, masters/postgraduate level

1.1 Introduction to Software Engineering (2011) – Lectured in English

In 2011 (spring), I was the teacher in charge of Introduction to Software Engineering course. This course is the first software engineering course in our curricula (taken by 2nd year bachelor students) and this year it has roughly 167 students. My responsibilities include selecting the educational material for the course, preparing and having lectures, organizing visiting lecturers, designing the course assignments and exams. I also have a team of four teaching assistants who help in taking care of the course. Development plan for this course is described in more details in this document (See Section 2).

1.2 Software Testing and Quality Assurance (2008-2010) – Lectured in English

In 2008-2010 (fall), I was the teacher in charge of Software Testing and Quality Assurance course. This is a master level course with roughly 25 students every year. During the past three years, I have made several changes to the course and they are described in more details in this document (See Section 2).

1.3 Seminar in Software Engineering (2003, 2006)

The seminar course has varying content and the teacher's main responsibilities were introductory lectures, helping the students to write good seminar papers, and organizing the seminar presentation day. The topic in 2006 was component based software engineering and in 2003 it was software evolution, bad code smells and anti-patterns. The course concluded with a seminar day that was organized in similar fashion to scientific conferences, i.e. students gave presentations and the floor was open for questions and discussion.

[1] https://noppa.tkk.fi/noppa/kurssi/t-76.3601/
[2] ECTS is European Credit Transfer and Accumulation System and 1 ECTS credit is roughly 27 hours of work
[3] https://noppa.tkk.fi/noppa/kurssi/t-76.5613
[4] https://noppa.tkk.fi/noppa/kurssi/t-76.4115/
[5] https://noppa.tkk.fi/noppa/kurssi/t-76.5650
[6] https://noppa.tkk.fi/noppa/kurssi/t-76.5612

To stimulate the discussion each paper had two designated student opponents who gave comments on the paper and the presentation.

1.4 Software Development Project (2001-2003, 2005-2007)

In software project course student groups of roughly 7 students complete a "real" software development project worth of 1500h for an external customer. I acted as teaching assistant (called "mentor" in this course). Each mentor helps 2-3 groups in using the software engineering methods required in the course, e.g. requirement engineering, systematic software testing, version management. Mentor works closely with the groups and there were roughly six mentor group meetings during the course. In addition feedback and tips were provided by e-mail.

2 Pedagogical studies and development projects

I completed the basic course of university teacher education in our university in the fall of 2008 (5 ECTS[7]). I am also currently participating in two other university teacher education courses (10 ECTS and 5 ECTS).

2.1 Pedagogical course: Introduction to University Teaching (5 ECTS) Improvements to: Software Testing and Quality Assurance course (5 ECTS)

This section describes the changes done to Software Testing and Quality Assurance course based on my participation of the Introduction to University Teaching course (5 ECTS).

I completed the introductory course of university teacher education in Aalto university in the fall of 2008 (5 ECTS[8]). The ideas from the course helped me to develop my own teaching of the Software Testing and Quality Assurance course whose responsible teacher I was between 2008(fall)-2010 (fall). Next I will describe the major changes between from the year 2008 to the year 2010.

2.1.1 Course in 2008

In 2008, the course consisted of 13 lectures, a group exercise, and a final exam. The lecture topics were not interconnected and the lectures had no lecture exercises that would have initiated interaction with the students. The group exercise had three phases and it consisted of software test design, review of the software test design, test execution, and test automation. The final exam at the end that was based on the lectures and the course book, and it tried to assess the learning in the traditional way final exams often do which led to students memorizing facts rather than focusing a deeper learning approach, i.e. students fill their heads with information and then "vomit" this to the exam paper and quickly forgot what they have memorized. Thus, in 2008 the course followed the behavioral style of learning.

2.1.2 Course in 2010

In 2010, the course had 10 lectures, 5 individual essays, and a group exercise. The course was organized under five themes: test management, testing techniques, software reviews, test automation and software quality. For each theme there was one overview lecture given by the course teacher and one visiting lecture. Lecture participation counted 12.5% of the grade. In the lectures given by the course teacher the lectures were made interactive by having the students create and present posters during the lectures (in two out of five lectures). During those lectures, the lecture time was often exceeded as the student were

[7] ECTS is European Credit Transfer and Accumulation System and 1 ECTS credit is roughly 27 hours of work
[8] ECTS is European Credit Transfer and Accumulation System and 1 ECTS credit is roughly 27 hours of work

engaged in discussing the posters. The visiting lecturers were specialists in the area they were lecturing about:

- In test management theme Maaret Pyhäjärvi who is test manager in industry (and former teacher of this course) talked about test management and her industrial experiences
- In testing techniques theme Juha Itkonen presented of his PhD research on exploratory testing
- In reviews theme Marko Komsi talked about the rise and fall of the reviews in a software company (F-Secure) and the reasons behind it
- In automated testing theme Mika Katara presented the work of his research group on model based testing of mobile phones
- In software quality theme Timo Lehtinen talked about his PhD research on Root Cause Analysis

The final exam was replaced by five individual essays. For each course theme, there was an essay that was based on the two lectures of the theme and few given articles. The essays made up 62.5% of the course grade. In the essay grading, the deep learning approach was encouraged as points were given an analytical analysis of the material, the comparison of the different viewpoints of the material, connecting the topic with past experiences, reflections, and grounded opinions. I see that through the evaluation of the essays a much better assessment of the learning was achieved in comparison to the final exam that was used in 2008. The students were given an immediate feedback of the essays and the essay scores also improved throughout the course (1st essay had average of 2.9 points, while 5th essay had average of 4.1 points). I believe this indicates the students learned during the course and gained knowledge not only of the course topics, but of analyzing material and presenting their ideas with good arguments.

The group exercise consisted of three phases: 1) Test design of test case based testing, exploratory testing and automated testing 2) Inspections of test design, 3) Test execution and implementation of automated testing. In 2010, the automation was done with Model-Based Testing (MBT) approach that is the current state of the art. In 2008, a scripted testing was used. With MBT, it made sense to include automation design with other types of test design. In comparison to 2008, the test automation was a separate phase.

2.1.3 Comparison of student feedback

Table 1 shows the comparison of numeric student feedback. The use of constructive learning methods has increased the students' overall assessment course by just a small margin. However, the big change is in the distribution of the students' overall assessment of the course. It could be stated that in 2008 everyone thought the course was OK, but in 2010 it brings up stronger emotions, e.g. the course is perceived either as Great or Terrible. The written feedback of the course supports this viewpoint:
Samples of Positive comments (2010):

- *I learned a lot, and pretty much everything I learned is directly useful in my future studies and career. I will recommend this course for other students.*
- *Good lectures. Set of articles (used in the course) make up more interesting material than the book.*
- *Writing essays really made me learn the theoretical part*
- *I liked the fact that there was no final exam and that essays were used to evaluate learning*
- *Visiting lectures were good*
- *Lecture exercises were a good idea*

and samples of negative comments

- *The instructions were too vague and just terrible. It does not make university a vocational school if the instructions are precise*
- *The essay questions were sometimes too vague*
- *Lectures were worse than average in this school*
- *Some visiting lectures were boring*

- *Practical arrangements were not working and some instructions come too late*

Table 1 Student feedback 2008 and 2010

	2008	2010	Interpretation of the data
Students' overall assessment of the course (scale 1-5, higher numbers are better)	avg:3.0 stdev:0.66	avg:3.1 stdev:1.22	From the students' assessment perspective, the development has not resulted in big improvement. Looking at behind the average reveals that the distribution in the students' assessment has increased greatly from 2008 to 2010.
Distribution of the students' overall assessment of the course	1-3.8 % 2-7.7 % 3-73.1 % 4-15.4 % 5-0.0 %	1-11.8 % 2-17.6 % 3-29.4 % 4-29.4 % 5-11.8 %	The course is now perceived by the students as much more liked and disliked. I believe that in 2008 the course was just like other courses in our university, but in 2010 it was considerable different leading to disagreement between students
The perceived workload in comparison to course credits (scale 1-5: 1- too much work, 3-workload just right, 5- too little work)	avg:2.9 stdev:0.46	avg:2.7 stdev:0.92	The perceived workload has increased slightly, but like in the students' overall assessment of the course the distribution in the perceived workload has increased.
The perceived difficulty of the course (scale 1-5: 1-too easy, 3 just right, 5-too difficult)	avg:3.0 stdev:0.00	avg:3.4 stdev:0.49	In 2008 the course was perceived just right in terms of difficulty. In 2010 one third of the students found the course to be somewhat too difficult. This change is in line with the other results.

2.1.4 Summary of Changes and Avenues for Further Improvement

Below I have summarized the big changes between 2008 and 2010. I also give some comments on these topics:

- Essays emphasizing the deep learning approach replaced the final course exam. Based on the student feedback and discussions (from 2010 and 2009) the essays were the best part of the course.
- Organization of the course topics under five main themes. Better high level organization makes the course contents more manageable for the students and for the teacher.
- Visiting lectures to give a wider perspective of the topics. Visiting lectures should be used in the future when the contents fit the courses learning goals.
- More interactive lectures, e.g. posters created and presented during the lectures. These need to have more emphasis in the future. Even though students may not like them it still makes them learn more than traditional lectures.
- Model based testing replaced scripted testing in the group exercise. University needs to teach the state of the art. Affecting the students' ideas of test automation will in the long run have a positive impact on the software industry as the students start to work in the industry.

The most notable avenue for further improvement is to work with the students who dislike the course. It is not good that almost 30% of the students assess the course lower than 3. This could be improved in various ways. For example some students had considerable difficulty coming up with good essays. Such students should be offered some more support. For example, a part of the lectures could be devoted to discussing the essays with the students who have difficulties. Furthermore, the reasons for having somewhat vague exercises needs to be better explained , but also the instructions needs to be clarified and the unintentional vagueness needs to be removed. Naturally, the minor problems with the practical arrangements need to be sorted out as well.

2.2 Pedagogical course: Course Clinic (5 ECTS) Improvements to: Introduction to Software Engineering course (5 ECTS)

In the spring of 2011, I participated in a teacher education course called Course Clinic (5ECTS). The goal of Course Clinic course is to improve existing courses or develop new ones. The course I improved was Introduction to Software Engineering course. Introduction to Software Engineering is a challenging course as it has almost two hundred students each year. Furthermore, it in principle should cover all the major parts of software engineering making it rather broad but shallow in terms of study contents. In the spring of 2011, I was also for the first time the teacher in charge of the Introduction to Software Engineering Course.

2.2.1 The course in 2010

Previously, the course has been a traditional lecture course with 20 lectures (10 given by the teacher in charge and other 10 given by the visiting lecturers). The idea behind the lecture structure has been that in the first ten lectures the course topics are covered according to the course book. Then, the course has been graded based on the final examination. The students also had the option of writing lecture summaries for the visiting lectures which would give in bonus points for the second midterm.

2.2.2 Development plan for 2012

Here I will describe the development plan for 2012, I have already used parts of the plan in 2011 when I lectured the course for the first time. The goals behind the plan are to follow the constructive learning approach and engage students in deep learning through exercises rather than focusing on the superficial memorization of the facts for the final exam.

2.2.3 Teaching and learning methods

First, I wish to reduce the number of lectures to make space for exercises. Thus, the number of lectures would be reduced from 20 to 12. All the lectures would be based on the course book and given by the teacher in charge. Each lecture would cover one course book chapter. This means that visiting lectures would be cut from the course contents. This is unfortunate but necessary to get time for student presentations.

Second, student presentations and interactive class room sessions would replace the final exam. The student presentation (10min) would take place in small exercise groups of roughly 25 students. After the presentation, there would be a short questions and answers session (5min) just like in scientific conferences. The presentation topics would come from Software Engineering book by Ian Sommerville[9]. In the book, each chapter concludes with set exercises, e.g. exercise 3.3. "*When would you recommend against the use of an agile method for developing a software system?*". Each of the exercises would be used as a presentation topic. As there are almost 200 students in the course we need 8 exercise groups that would be headed by teaching assistants. I believe that it is even better that teacher in charge would not be present in the exercise groups as it would change the group dynamics and would likely reduce the interaction between the students as the majority of the students would be looking at the teacher to provide the correct answer. From the students' viewpoint, there would be one exercise group session for each lecture.

[9] Sommerville I., *Software Engineering*, 9th edition, Pearson, pp. 1-773

Third, lecture diaries would replace lecture summaries. Furthermore, the lecture diaries would be mandatory part of the course. There is subtle but important difference between lecture diaries and summaries. While both should contain the main points of the lecture and analysis of the lecture topics, lecture diaries should be more personal and contain things like reflecting the lecture topics to past learnings. In lecture diaries there would also be opinions of the lecture topic that would typically be missing from lecture summaries. Lecture diaries were part of the course in 2011, but they were not mandatory although 40% of the students' course grade was based on them. Brief feedback will be given based on each submitted diary entry.

Fourth, students would use Peerwise system[10] to create multiple choice questions and answer multiple choice questions created by other students. Peerwise allows providing explanations for the correct answer, commenting on the questions, and rating the questions by other students. I believe that creating high quality multiple choice questions encourages the deep learning approach and that answering multiple choice questions helps one in getting a good overview of the course contents.

2.2.4 Grading and Quality Control of Learning

This section explains how the students will be graded. Grading has huge impact on how the students work in the course. Therefore, it is important to align the grading with the course activities. The principles behind the grading are

1) More is better, e.g. two presentations are better than one
2) Higher quality is better, e.g. one high quality lecture diary entry is better than one medium quality diary entry.

The course grade is formed as follows:

- Course grade = 0.1*PC + 0.1*PA + 0.4*L + 0.4*S
- S = Student presentation grade
- L = Lecture diary grade
- PC = Peerwise multiple choice question creation grade
- PA = Peerwise multiple choice question creation grade

For student presentations (S) we can only utilize principle one (more is better). This is due to the fact that we cannot guarantee that all students who have prepared a presentation would have chance to present it. In fact, in two hour group exercise session only 6 students out of 25 will have a presentation. The students will get a presentation point each time they are present in an exercise session and have submitted presentation outline in advance. The teaching assistant will at the exercise session randomly select the ones that will actually be giving their presentations. The S in above formula will be graded as follows: 4 presentation points give one as the presentation grade (S=1), 5 points will result in the presentation grade of 2 (S=2), 6 points will give 3 (S=3), 7 points will give 4 (S=4), and 8 points or more will give (S=5).

For lecture diaries (L) both principles are utilized (more is better and higher quality is better. The lecture diary entries will be graded with a grading scheme from 0 to 3 scale (3 – excellent, 2 – OK, 1 – major shortcomings, 0 – disaster). All students should score 2 points when devoting two hours of effort. The L in the above formula is created as follows: 8 points L=1, 10 point L=2, 12 points L=3, 14 points L=4, 16 points L = 5.

[10] http://peerwise.cs.auckland.ac.nz/

For Peerwise question creation (PC) grading both principles are utilized. Each Peerwise question is rated by fellow students from 1-5 scale, and thus each question will score as many points as it the average rating given by fellow students. 12 points PC =1, 15 points PC=2, 18 points PC = 3, 21 points PC = 4, 24 point PC = 5

For Peerwise question answering (PA) grading both principles are used. In grading only the number of correct answer is counted, but there is no limit to the number questions you can answer. 20 correct answers PA =1, 25 correct answers PA =2, 30 correct answers PA=3, 35 correct answers PA=4, 40 correct answers PA=5

2.2.5 Feasibility of the plan

Based on the realized effort of the teaching staff in 2011, I created an estimate of the teaching staff effort (= course budget) for the course plan of 2012. I presented this plan for our vice head of department professor Tomi Männistö and he saw the effort as acceptable considering the high number of students.

Here I present a summary of the effort estimates of the teaching staff (detailed calculation will be provided on request)

- Each lecture (2h) takes 10h (includes proper preparation) -> 12*10h = 120h
- Each exercise group takes 2h (preparation) and 2h (the exercise session itself) -> 4h*8*12 = 384h
- Grading each lecture diary entry takes 15minutes (based on 2011 data) -> 15min*12*100= 300h
- Peerwise grading is rather straight forward and should take not more than 20h in total

I have also calculated the students' workload estimates and they are in line with the given credits (detailed calculations will be provided on request).

3 Teaching philosophy

I think that research is more exciting, but teaching brings greater fulfillment. Thus, it would be pointless to set one before the other. In the fall of 2008, I completed the basic course of university teacher education (5 ECTS[11]) in the Helsinki University of Technology. This course had big impact on me and what I think about teaching, the role of the teacher and the role of the student. Before the course, I viewed teaching simply as something that interfered with research. During the course*, I realized that teaching is highly important in universities and it truly matters how the teaching carried out*.

I see students as junior members of academic community rather than customers. For a long time I viewed students as customers as do many professors around the world. However, during 2011 I attended a teacher education where the role of the student was discussed and during the course one of our instructors challenged the view 'students as customers' and proposed the aforementioned view 'students as junior members of academic community'. To me this alternative view first sounded rather poor. However, after thinking and reflecting about this I realized that the instructor's view had been a correct one. Below I have highlighted some problem with "students as customers" view.

- In the customer-provider relationship, both parties typically try to achieve the local maximum rather than common good. Thus, students would shop for courses that they see immediately valuable in the job market or easy to pass to maximize the credits they receive. Students would also act as irresponsibly as customers often think that provider exits for solely to fulfill the customers'

[11] ECTS is European Credit Transfer and Accumulation System and 1 ECTS credit is roughly 27 hours of work

needs. Teachers on the other would be trying to optimize efficiency and edutainment[12] to maximize their production of credits. Teachers would also be more willing to make courses easier to please the customers.

- In the junior-senior relationship, the relationship is based on trust and responsibility. Students act responsibly and trust that the work they put in will eventually pay off even there are no immediate benefits visible. Teachers on the other hand understand their responsibility to the discipline they represent, and to the society and state they are working for.

I see teachers as gardeners that specialize in fostering plants with strong, versatile and deep roots. With these deep roots, students are available to cherish in whatever environment they happen to be in. Teachers face somewhat similar difficulty that the gardeners have. Like the garners cannot see the roots of the plants they are growing, similarly the teachers cannot see the brains of the students they are teaching. They must both rely on similar methods, e.g. taking care of plants/students and put them on the right kind of soil where they can grow their roots/brain.

Building on my views of students and teachers, I strongly believe that constructive learning methods that emphasize deep learning are ones that should be used in the university environment. As a teacher, I believe that activating students and giving them challenging and rewarding assignments results in the best learning outcomes. Some of the learning methods I have utilized in my courses are:

- Lectures with exercises and discussions
- Essay which are graded based on the understanding about the topic and reflection past leaning
- Students creating and answering Peerwise questions
- Student presentations
- Students opponing other students
- Technical exercises done in groups, e.g. create test designs and use Graphwalker tool to automate testing

[12] The reader is encouraged to think how the word *edutainment* is formed from the words of education and entertainment.