UNIVERSITÄT
BERN

Vizerektorat Lehre, iLUB, Hochschulstrasse 6, 3012 Bern

Persönlich/Vertraulich
Prof. Dr. Oscar Nierstrasz
Institut für Informatik und angewandte Mathematik
Neubrückstr. 10
3012 Bern

Vizerektorat Lehre

Lehrveranstaltungsevaluation

Report of evaluation: HS18 Einführung in Software Engineering (2420)

Dear Mr./Mrs. Prof. Dr. Nierstrasz

Please find here the results of the evaluation of your course "Einführung in Software Engineering". Following the scanning of the questionnaires, this report was automatically generated and mailed to you.

The questionnaire used was appropriate to the course type Vorlesung. In the report, you first see the mean values of the most important dimensions:

- Conveying the course content
- Course materials to assist Learning
- Commitment of the lecturer
- Complexity and Scope
- Assessment of Individual Lectures

In the second part of the report, you see the answers to all the questions. The number of answers, the mean value and the values differing from it are also given.

Grade 1 on the left hand side equals the lowest grade given by the students, grade 5 or more on the right hand side the highest grade. In 'complexity and scope' grade 3 corresponds to 'exactly right' and is therefore the best grade. In the overall assessment of the course, grade 6 means the best result.

The free comments at the end of the questionnaire are only read by the lecturer him/herself and won't be evaluated statistically. Please don't pay much attention to negative statements of single persons. You are to look closely in case of frequent occurence of similar comments.

Please briefly discuss the results with your students before the end of the semester. You will find a presentation template on the last pages of the report. By giving serious consideration to the feedback of the students, you can contribute to higher future response rate.

In case you wish to learn more about how to improve your teaching, you might want to discuss the results with the staff of the 'Hochschuldidaktik' (mail address: hd@zuw.unibe.ch). Please bring a copy of the report with you, since the staff of Hochschuldidaktik do not have access to evaluation results.

You might find guidelines, regulations, and information about the process under www.lehrveranstaltungsevaluation.unibe.ch (documents in German).

Should you need more information, you may also contact us by e-mail.

Yours sincerely

D. Wuillemin
Evaluation office
Vice-rectorate of teaching

Daniela Wuillemin
Supportstelle für ICT-gestützte Lehre und Forschung (iLUB)
Hochschulstrasse 6, 3012 Bern

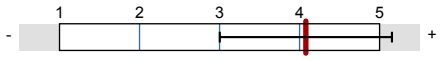
Tel. +41 31 631 51 07
lehrevaluation@ilub.unibe.ch
www.lehrveranstaltungsevaluation.unibe.ch

Prof. Dr. Oscar Nierstrasz

HS18 Einführung in Software Engineering (2420)
No. of responses = 28

Overall indicators

1. Conveying the course content (α = 0.91)



av.=4.08
dev.=1.08

2. Course materials to assist Learning (α = 0.91)

av.=4.07
dev.=1.24

3. Commitment of the lecturer (α = 0.75)

av.=4.74
dev.=0.56

4. Complexity and Scope (α = 0.49)

av.=3.49
dev.=0.76

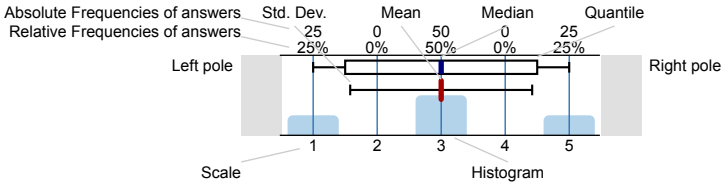
8. Assessment of Individual Lectures (α = 0.87)

av.=3.8
dev.=1.1

Survey Results

Legend

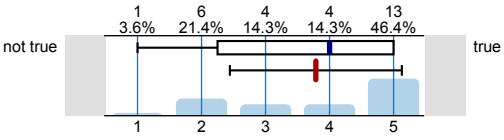
Question text



n=No. of responses
av.=Mean
md=Median
dev.=Std. Dev.
ab.=Abstention

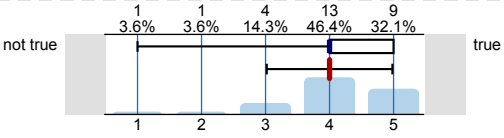
1. Conveying the course content

1.1) The course follows a coherent structure.



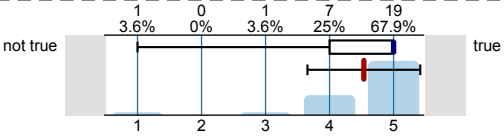
n=28
av.=3.79
md=4
dev.=1.34

1.2) The wider context of the subject matter is sufficiently elucidated.



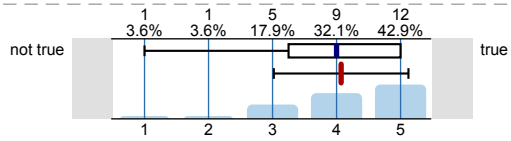
n=28
av.=4
md=4
dev.=0.98

1.3) The lecturer expresses him-/herself clearly and comprehensibly.



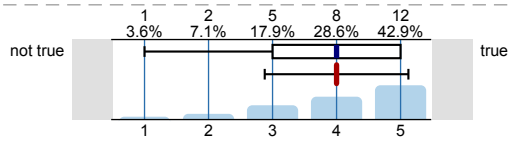
n=28
av.=4.54
md=5
dev.=0.88

1.4) The course provides an adequate overview of the subject matter treated.



n=28
av.=4.07
md=4
dev.=1.05

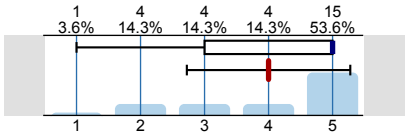
1.5) The design of the course contributes to an understanding of the subject matter.



n=28
av.=4
md=4
dev.=1.12

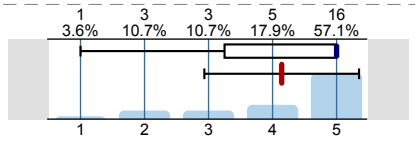
2. Course materials to assist Learning

2.1) There is overall enough material provided to assist the learning process (slides, coursematerial, hand-outs, etc.).



n=28
av.=4
md=5
dev.=1.28

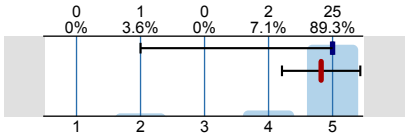
2.2) The course materials (slides, course manuals,hand-outs, etc.) are overall of sufficient quality.



n=28
av.=4.14
md=5
dev.=1.21

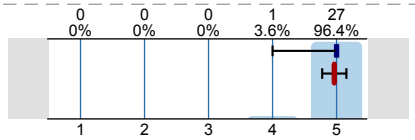
3. Commitment of the lecturer

3.1) The lecturer takes students seriously.



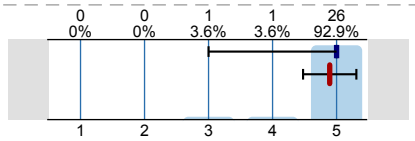
n=28
av.=4.82
md=5
dev.=0.61

3.2) The lecturer is friendly and respectful towards students.



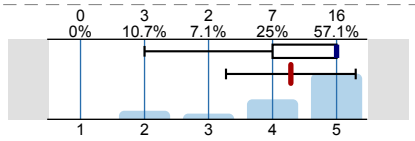
n=28
av.=4.96
md=5
dev.=0.19

3.3) The lecturer addresses questions and suggestions from students adequately.



n=28
av.=4.89
md=5
dev.=0.42

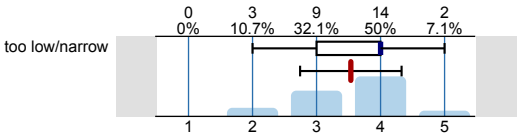
3.4) The lecturer seems to care about his/her students' learning progress.



n=28
av.=4.29
md=5
dev.=1.01

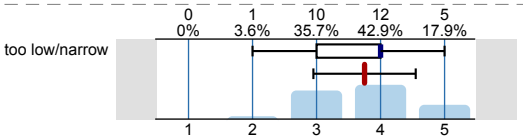
4. Complexity and Scope

4.1) The degree of difficulty of the course is:



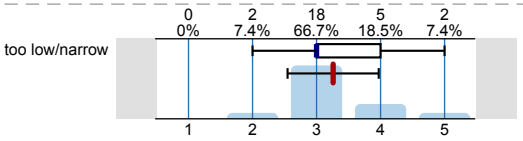
n=28
av.=3.54
md=4
dev.=0.79

4.2) The amount of content of the course is:



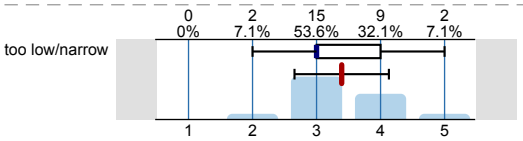
n=28
av.=3.75
md=4
dev.=0.8

4.3) The pace of the course is:



n=27
av.=3.26
md=3
dev.=0.71

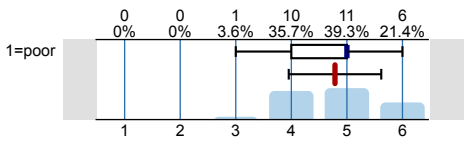
4.4) The amount of knowledge presupposed by the course is:



n=28
av.=3.39
md=3
dev.=0.74

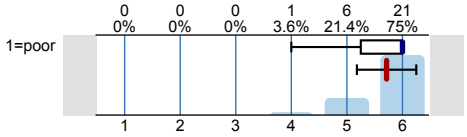
5. Overall Assessment

5.1) How would you grade the course as a whole?



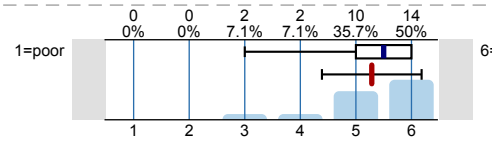
n=28
av.=4.79
md=5
dev.=0.83

5.2) How would you grade the lecturer with regard to subject expertise?



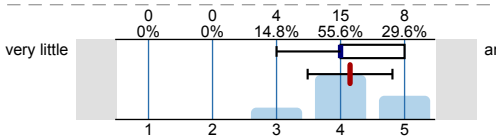
n=28
av.=5.71
md=6
dev.=0.53

5.3) How would you grade the lecturer with regard to teaching methods?



n=28
av.=5.29
md=5.5
dev.=0.9

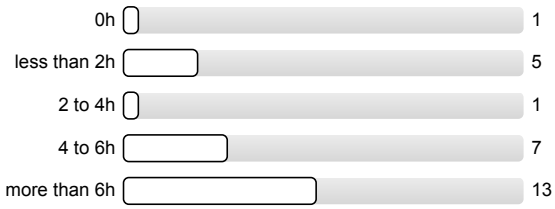
5.4) The course has taught me:



n=27
av.=4.15
md=4
dev.=0.66

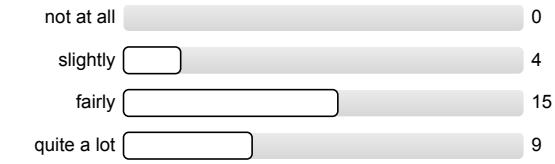
6. Socio-demographic Data and Background Variables

6.1) How many hours per week did you invest in preparation and revision for the course (on average)?



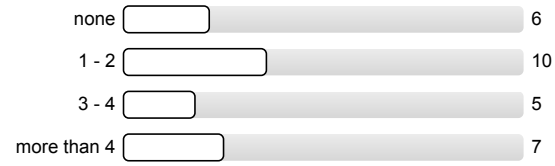
n=27

6.2) Was the topic of interest to you?



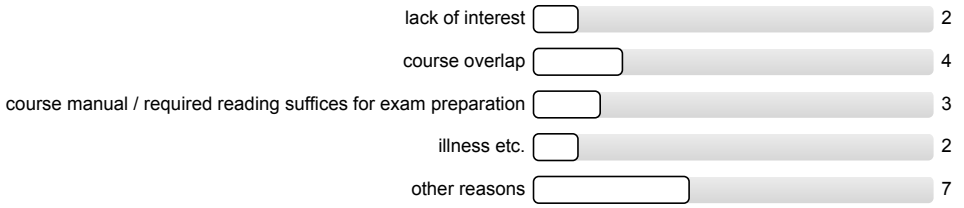
n=28

6.3) How many lectures did you miss?



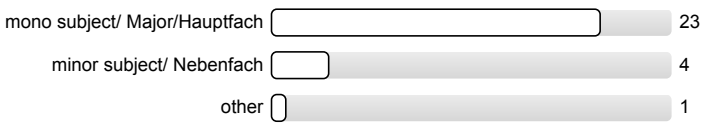
n=28

6.4) If you missed more than 2 lectures, please give one reason:



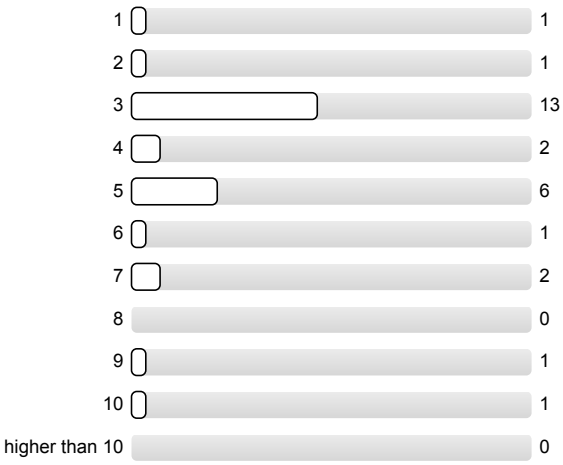
n=18

6.5) Allocation of the course in your study programme:



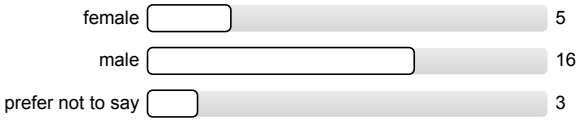
n=28

6.6) Your current number of semesters since starting your studies:



n=28

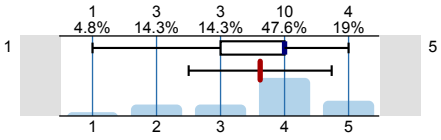
6.7) Sex:



n=24

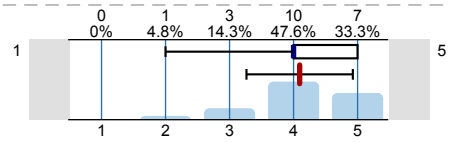
8. Assessment of Individual Lectures

8.1) **Introduction: The Software Lifecycle**



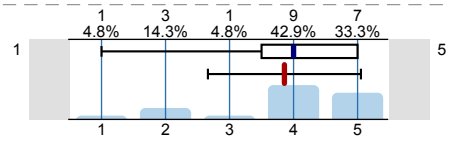
n=21
av.=3.62
md=4
dev.=1.12

8.2) **Requirements Collection**



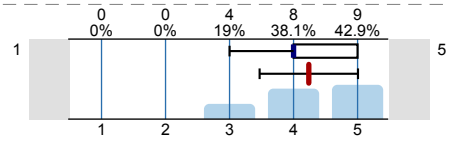
n=21
av.=4.1
md=4
dev.=0.83

8.3) **Agile Methods (Pietari Kettunen)**



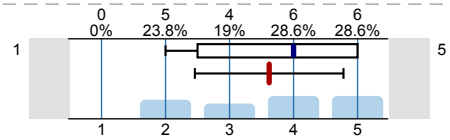
n=21
av.=3.86
md=4
dev.=1.2

8.4) **Responsibility-Driven Design**



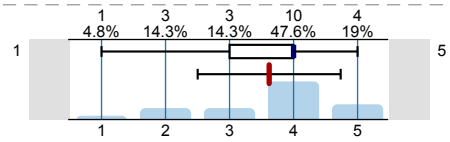
n=21
av.=4.24
md=4
dev.=0.77

8.5) **Modeling Objects and Classes**



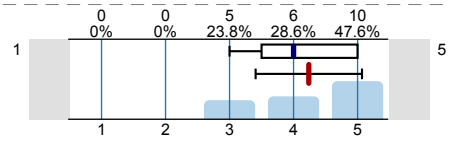
n=21
av.=3.62
md=4
dev.=1.16

8.6) **Modeling Behaviour**



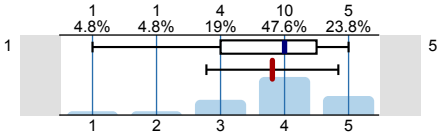
n=21
av.=3.62
md=4
dev.=1.12

8.7) **User Interface Design**



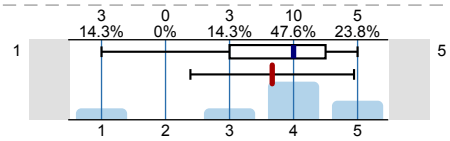
n=21
av.=4.24
md=4
dev.=0.83

8.8) **Software Quality**



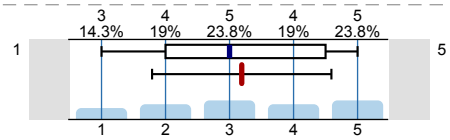
n=21
av.=3.81
md=4
dev.=1.03

8.9) **Guest lecture: Software Testing (Manuel Oriol)**



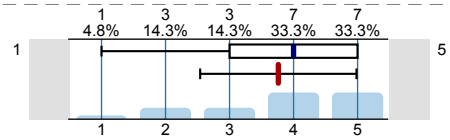
n=21
av.=3.67
md=4
dev.=1.28

8.10) **Software Security**



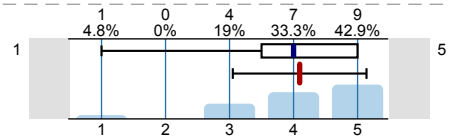
n=21
av.=3.19
md=3
dev.=1.4

8.11) **Software Metrics**



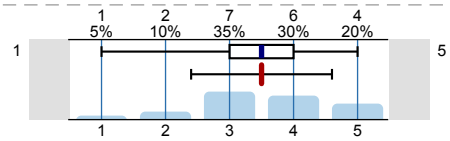
n=21
av.=3.76
md=4
dev.=1.22

8.12) **Guest lecture: Project Management (Jan Hornwall)**



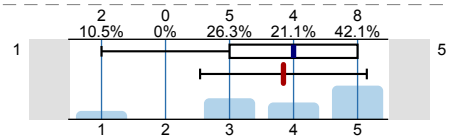
n=21
av.=4.1
md=4
dev.=1.04

8.13) **Software Architecture; Guest lecture: Software Architecture in practice (Erwann Wernli)**



n=20
av.=3.5
md=3.5
dev.=1.1

8.14) **Guest lecture: SE in practice (Peter Gfader)**



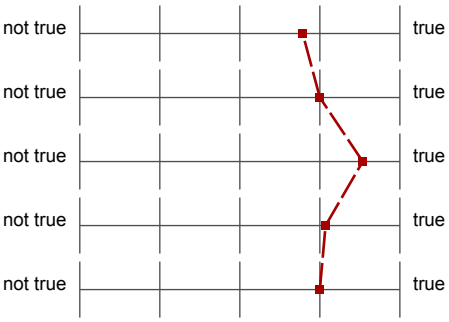
n=19
av.=3.84
md=4
dev.=1.3

Profile

Subunit: Phil.-nat. Fakultät
Name of the instructor: Prof. Dr. Oscar Nierstrasz
Name of the course: (Name of the survey) Einführung in Software Engineering

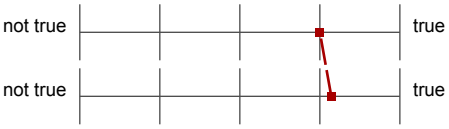
Values used in the profile line: Mean

1. Conveying the course content



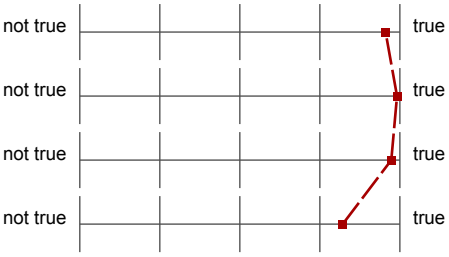
1.1)	The course follows a coherent structure.	not true	true	n=28	av.=3.79	md=4.00	dev.=1.34
1.2)	The wider context of the subject matter is sufficiently elucidated.	not true	true	n=28	av.=4.00	md=4.00	dev.=0.98
1.3)	The lecturer expresses him-/herself clearly and comprehensibly.	not true	true	n=28	av.=4.54	md=5.00	dev.=0.88
1.4)	The course provides an adequate overview of the subject matter treated.	not true	true	n=28	av.=4.07	md=4.00	dev.=1.05
1.5)	The design of the course contributes to an understanding of the subject matter.	not true	true	n=28	av.=4.00	md=4.00	dev.=1.12

2. Course materials to assist Learning



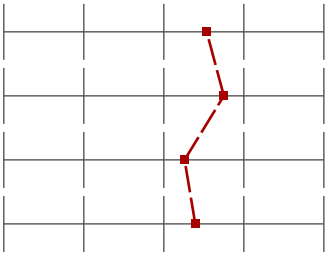
2.1)	There is overall enough material provided to assist the learning process (slides, coursematerial, hand-outs, etc.).	not true	true	n=28	av.=4.00	md=5.00	dev.=1.28
2.2)	The course materials (slides, course manuals, hand-outs, etc.) are overall of sufficient quality.	not true	true	n=28	av.=4.14	md=5.00	dev.=1.21

3. Commitment of the lecturer



3.1)	The lecturer takes students seriously.	not true	true	n=28	av.=4.82	md=5.00	dev.=0.61
3.2)	The lecturer is friendly and respectful towards students.	not true	true	n=28	av.=4.96	md=5.00	dev.=0.19
3.3)	The lecturer addresses questions and suggestions from students adequately.	not true	true	n=28	av.=4.89	md=5.00	dev.=0.42
3.4)	The lecturer seems to care about his/her students' learning progress.	not true	true	n=28	av.=4.29	md=5.00	dev.=1.01

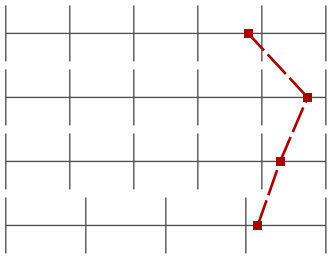
4. Complexity and Scope



4.1)	The degree of difficulty of the course is:	too low/narrow	too high / wide	n=28	av.=3.54	md=4.00	dev.=0.79
4.2)	The amount of content of the course is:	too low/narrow	too high / wide	n=28	av.=3.75	md=4.00	dev.=0.80
4.3)	The pace of the course is:	too low/narrow	too high / wide	n=27	av.=3.26	md=3.00	dev.=0.71
4.4)	The amount of knowledge presupposed by the course is:	too low/narrow	too high / wide	n=28	av.=3.39	md=3.00	dev.=0.74

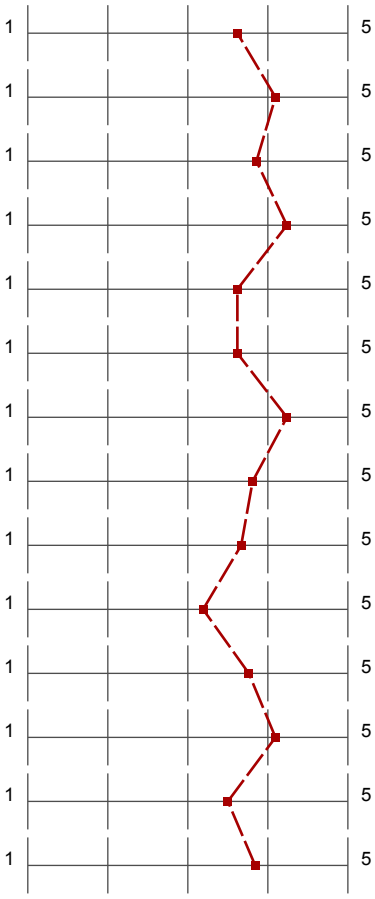
5. Overall Assessment

5.1)	How would you grade the course as a whole?	1=poor	6=excellent	n=28	av.=4.79	md=5.00	dev.=0.83
5.2)	How would you grade the lecturer with regard to subject expertise?	1=poor	6=excellent	n=28	av.=5.71	md=6.00	dev.=0.53
5.3)	How would you grade the lecturer with regard to teaching methods?	1=poor	6=excellent	n=28	av.=5.29	md=5.50	dev.=0.90
5.4)	The course has taught me:	very little	an awful lot	n=27	av.=4.15	md=4.00	dev.=0.66



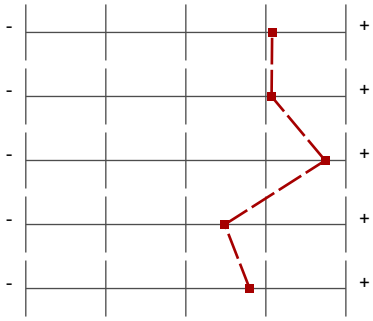
8. Assessment of Individual Lectures

8.1)	Introduction: The Software Lifecycle	1	5	n=21	av.=3.62	md=4.00	dev.=1.12
8.2)	Requirements Collection	1	5	n=21	av.=4.10	md=4.00	dev.=0.83
8.3)	Agile Methods (Pietari Kettunen)	1	5	n=21	av.=3.86	md=4.00	dev.=1.20
8.4)	Responsibility-Driven Design	1	5	n=21	av.=4.24	md=4.00	dev.=0.77
8.5)	Modeling Objects and Classes	1	5	n=21	av.=3.62	md=4.00	dev.=1.16
8.6)	Modeling Behaviour	1	5	n=21	av.=3.62	md=4.00	dev.=1.12
8.7)	User Interface Design	1	5	n=21	av.=4.24	md=4.00	dev.=0.83
8.8)	Software Quality	1	5	n=21	av.=3.81	md=4.00	dev.=1.03
8.9)	Guest lecture: Software Testing (Manuel Oriol)	1	5	n=21	av.=3.67	md=4.00	dev.=1.28
8.10)	Software Security	1	5	n=21	av.=3.19	md=3.00	dev.=1.40
8.11)	Software Metrics	1	5	n=21	av.=3.76	md=4.00	dev.=1.22
8.12)	Guest lecture: Project Management (Jan Hornwall)	1	5	n=21	av.=4.10	md=4.00	dev.=1.04
8.13)	Software Architecture; Guest lecture: Software Architecture in practice (Erwann Wernli)	1	5	n=20	av.=3.50	md=3.50	dev.=1.10
8.14)	Guest lecture: SE in practice (Peter Gfader)	1	5	n=19	av.=3.84	md=4.00	dev.=1.30



Profile Line for Indicators

Subunit: Phil.-nat. Fakultät
Name of the instructor: Prof. Dr. Oscar Nierstrasz
Name of the course: (Name of the survey) Einführung in Software Engineering



1. Conveying the course content	av.=4.08	dev.=1.08
2. Course materials to assist Learning	av.=4.07	dev.=1.24
3. Commitment of the lecturer	av.=4.74	dev.=0.56
4. Complexity and Scope	av.=3.49	dev.=0.76
8. Assessment of Individual Lectures	av.=3.80	dev.=1.10

Comments Report

7. Comments

7.1) What did you like about the course?

- Combination of theory and practical work. Guest lecture were helpful to get insight into IT-industry. Support from assistants.
- Guest lectures, project, good slides
- Interesting subject. Helpful information/methods
- Mr Nierstrasz is the best lecturer next to Mr Strahm at this institute. Was fun to attend this course. Also, the project we had to do was very motivational and we did learn a lot.
- Practical project over a whole semester. Podcasts.
- Project
- Project and guest lectures
- The guest lectures were funny / very cool / not repetitive
- The guest lectures, to see practitioners
- The hidden slides are helpful
- The lecturer has excellent expertise in both this subject and teaching methods. Many guest lectures with interesting topics.
- The prof has a really good teaching style.
- The project!
- Variety of topics covered, Podcast, range of ideas and book recommentations
- well structured information

7.2) What did you not like about the course?

- Guest lectures were good, but often review of material covered. Figure out the issue with the static, some of the diagram drawing was good practic but big time loss.
- Little cohesion between course and project
- Not enough information on how to handle/start the project
- The practical exercise started off with an extremely sleep learning curve. Ease into it a little bit more.
- There should be more credits for the amount of work. Project was too much work. By university regulations you are forced to make and present evaluation during the lecture!!!!
- There were too many topics.
- Work load was pretty high; since I did visit other courses and have a 80%-job, I could not profit from this course as much as I would have liked. But actually, that was my own choice…
- guest lectures (--> slides)
- not a lot of practice for the actual lecture; the description of the project was too open to know what to do and too fixed to decide what to do!
- not clear structure, exercises nothing to do with lecture, more UUL practice needed, more examples
- some terms are not clear. e.g. requirement analysis and analysis. Are they the same? So the slides are sometimes ambiguous.
- the exercise hours were mostly not really thought through and I think we were not given enough information about the project
- uncorrelation of project and lecture

7.3) Suggestions for improvements?

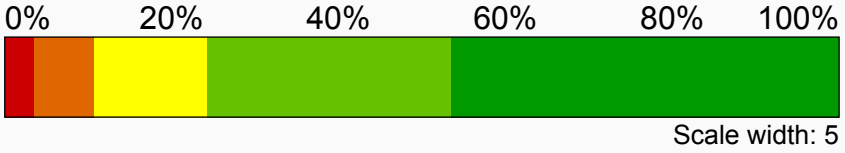
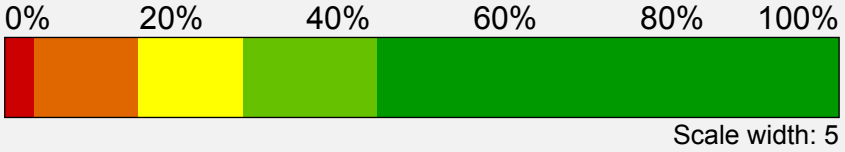
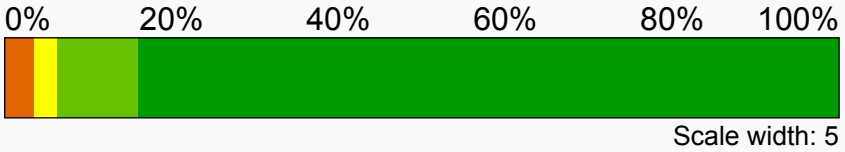
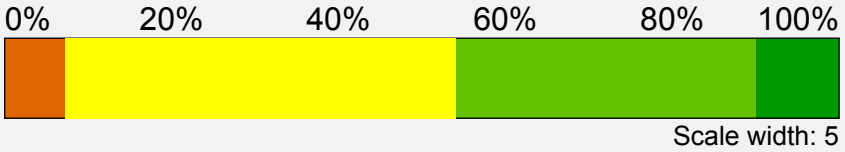
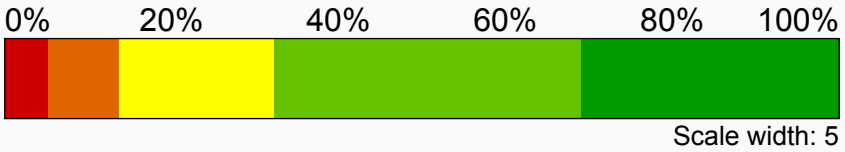
- A bit less of a "hands-off" approach by assistants in exercises.

- Cancel final exam. Do not make evaluation on exam day.
- Give exercises like drawing UML. Don't give a project description and scaffolding, only talks to project owner and start with zero (perhaps specify what tools to use).
- Have list of deliverables and project information all in one place. Give detailed information on how the grading is done (as it accounts for 40%).
- Make the lecture more about the project s.t. project and lecture actually work together
- Maybe some exam questions for preparation (old exams)
- More examples/interactive work for UML diagrams. I still don't understand … and they look awfully alike (?).
- More material for learning the programming languages. Very hard to pick it all up from scratch.
- Provide additional material for subjects only treated in the exercise lectures.
- Shorten the lecture, help more with the project
- more examples for complex topics like some UUL-stuff and architecture

Einführung in Software Engineering

Responses = 28 questionnaires

Prof. Dr. Oscar Nierstrasz

	- +	Ø	dev.
1. Conveying the course content	 Scale width: 5	4.08	1.08
2. Course materials to assist Learning	 Scale width: 5	4.07	1.24
3. Commitment of the lecturer	 Scale width: 5	4.74	0.56
4. Complexity and Scope Complexity and Scope: left pole=too low, right pole=too high; grade 3=exactly right	 Scale width: 5	3.49	0.76
8. Assessment of Individual Lectures	 Scale width: 5	3.8	1.1

dev.=Std. Dev.