

# Integration of a Student Project Into the Course ‘Software Testing’

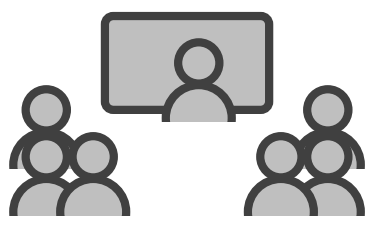
## Definition and Application of the TEPI Concept

Sascha Lehmann (E16 - STS), Ann-Christine Kyler (E15 - SVA)

### Course Facts

Participant count:	35 (initial), 29 (final)
Course Level:	Master
Course Period:	Summer Term, 2017
Course Weighting:	6 ECTS
Module Structure:	Lecture + Project Course (PBL with 5 phases)
Study Programs:	CS (Computer Science) ICS (Information and Communication Systems) IIW (Computer Science and Engineering)
Examination Method:	Graded Project Submissions and Interviews
Past Realizations:	1 (Summer Term 2016, Oral Exam)

### Problem Definition



The old course design led to students not being confident in their ability to apply testing methods in practice.

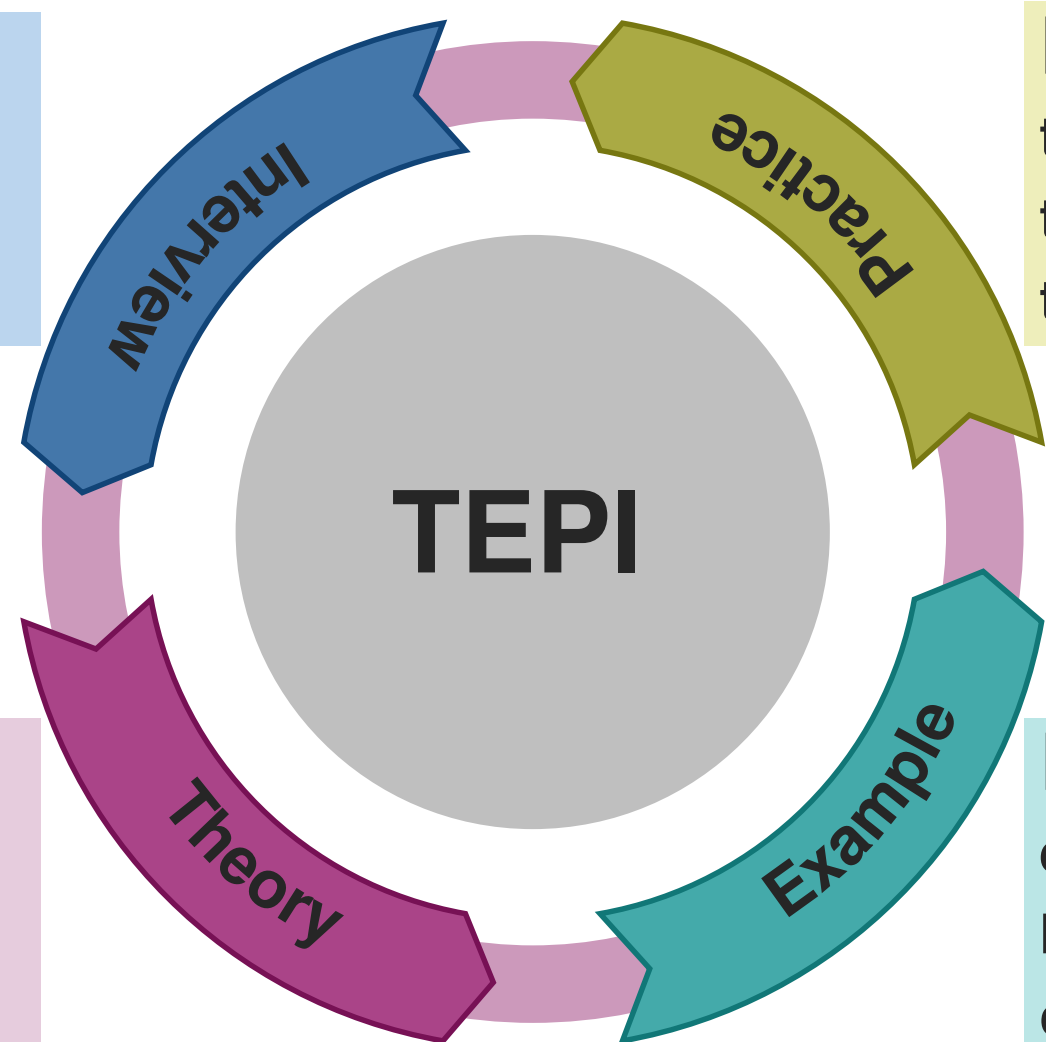
### Objective

Our goal is to provide practical software testing experience, while still teaching the theoretic backgrounds. After the course, the students should be more confident when they design and apply software tests.

### New Concept

**Interview** The project advisor tests and corrects the knowledge of the concepts in an interview.

**Theory** Theoretic concepts are presented to students in the lecture, and taken up in the first exercise of each phase.



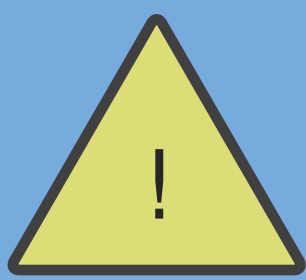
**Practice** Students apply the practiced concepts to their own project in the final task of each phase.

**Example** In the second exercise of a phase, students have to apply the theoretic concepts to a small example given in the exercise sheet.

The students should be able to apply the theoretical methods presented in the lecture. To achieve this, a student project is integrated into the module. As the underlying concept project based learning, similar to [1], is used. Instead of an exam the students are graded based their solution of tasks organised in five phases. The students work in groups, but submit individual written solutions and are interviewed on their theoretical knowledge and reasoning in each phase. To increase the motivation of the students, the groups can choose an open source project to apply the testing methods to. Hereby the students can also partake in a real software project.

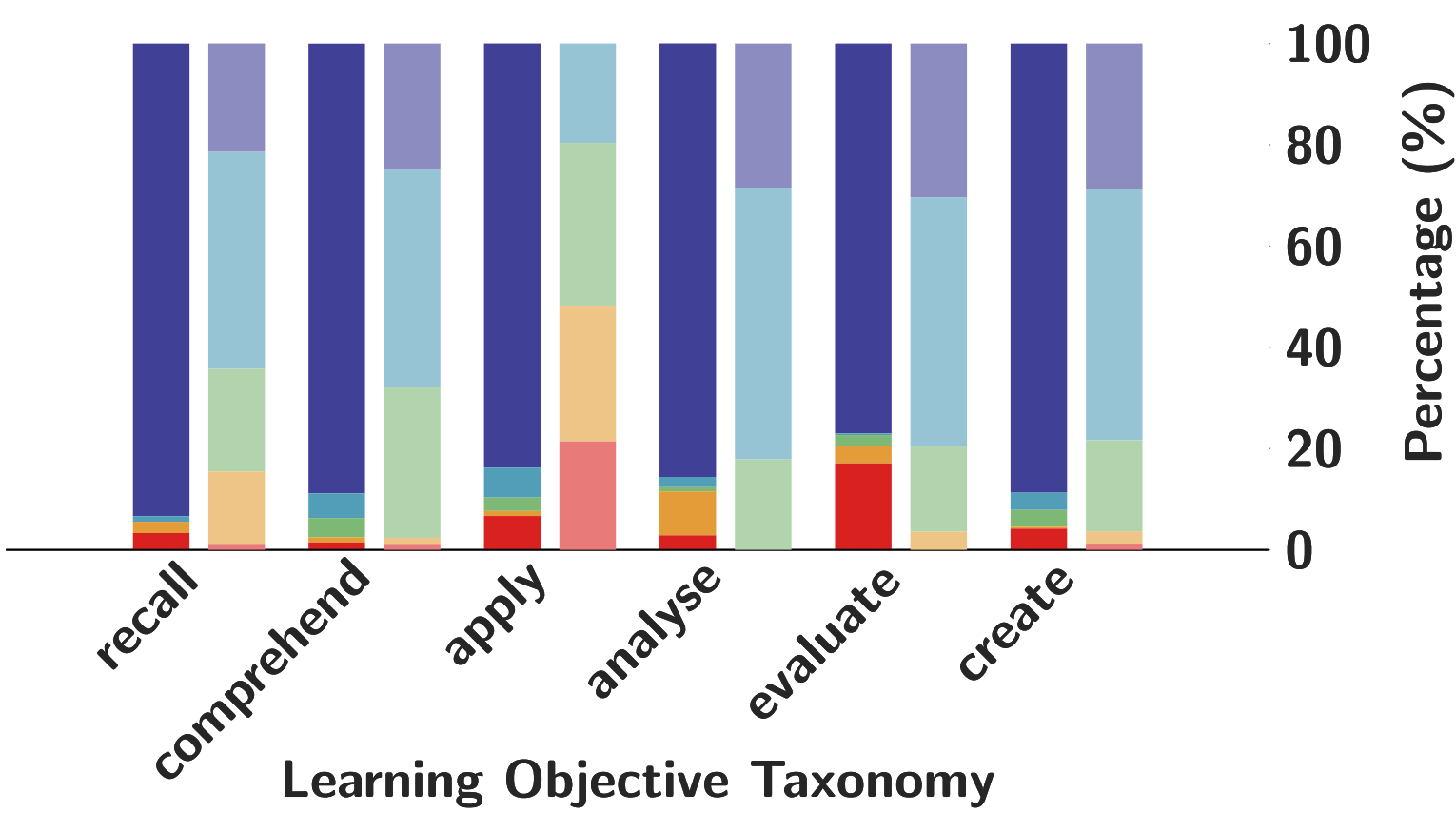
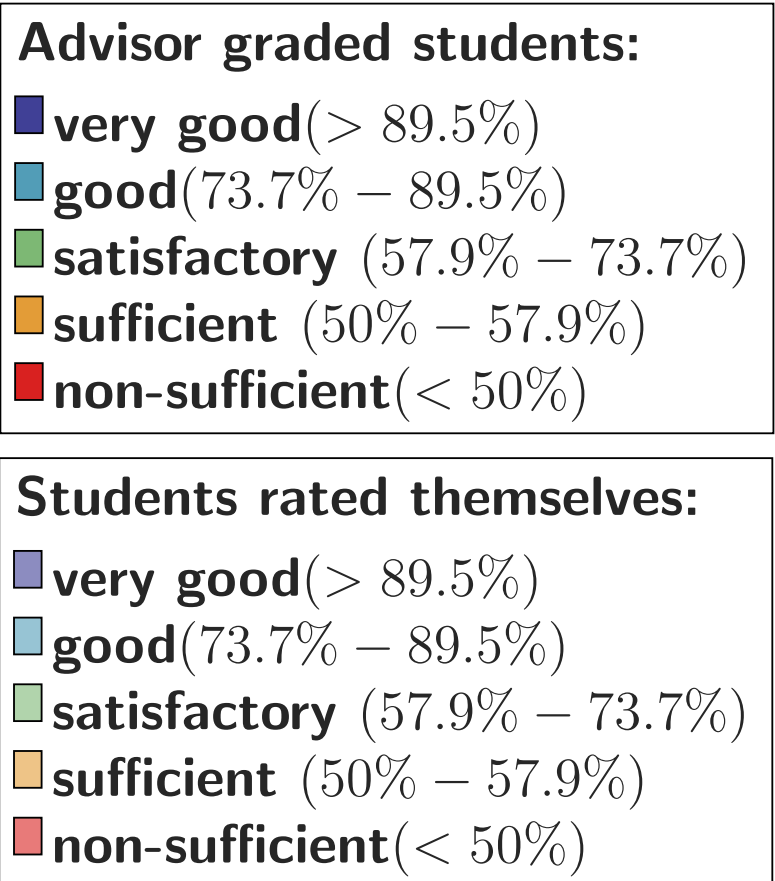
### Challenges

- Keep group size small while keeping supervising efforts small
- Reduce interview time while still being just and comparable
- Allowing degrees of freedom while keeping the project supervisable
- Connecting lecture and project sensibly

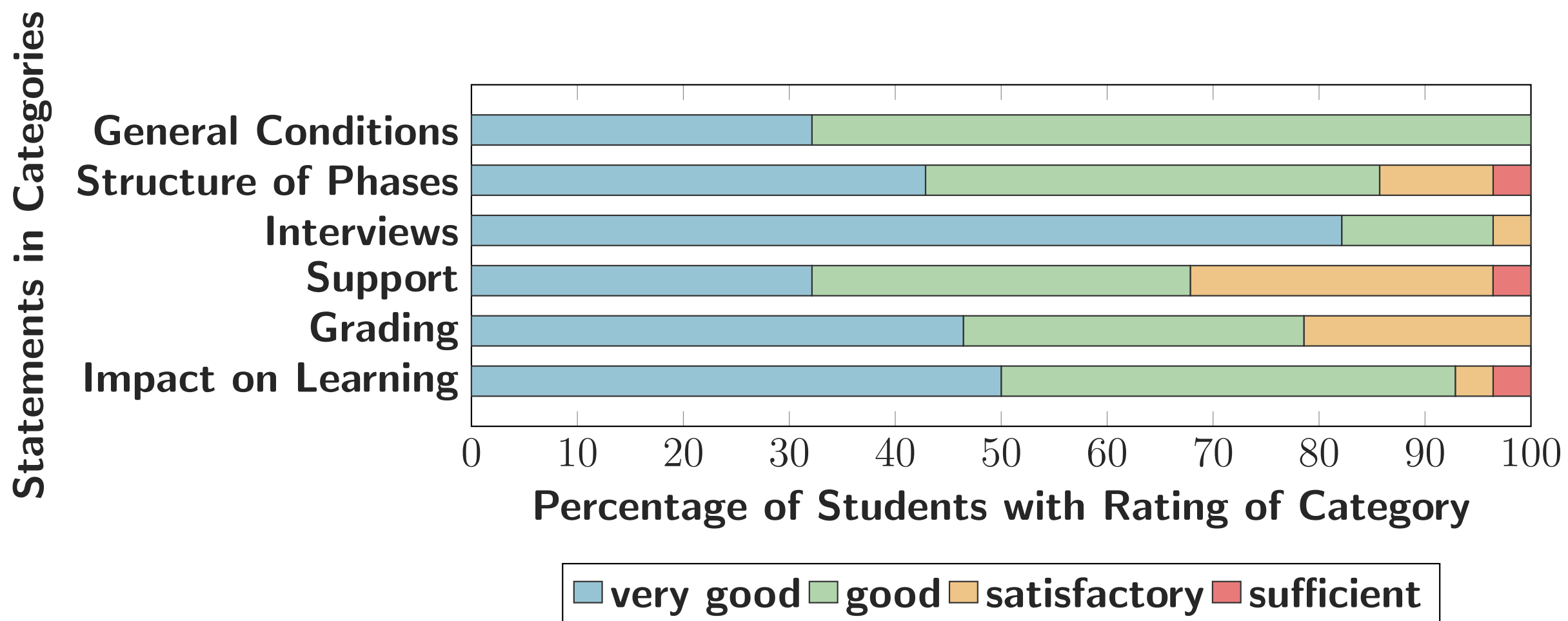


### Evaluation

To evaluate the success of the new concept, multiple sources of information where used: the results of the projects, as well as questionnaires. The main questionnaire allowed self assessment in the categories methodological skills, reflection and content knowledge and was based on [2]. These were mapped to the learning objective taxonomy as presented in [3].



The students were additionally handed a feedback sheet that was designed to support future course improvements. The questions concerned the project organisation, the grading fairness and the overall impression of the course.



"The most tasks could be solved alone. The aspect of group work is useful to discuss the solutions and problems."

"Lecture should be more ahead of the tasks in my opinion."

"The groups were quite large - smaller could've been better."

"It was sometimes not clear to us, how much our submissions must differ. That sometimes impeded our group work."

### Conclusion

"The interviews reinforced my long-term memorization of contents since I prepared the interviews."

"The group didn't do anything together besides choosing the project."

- +
- The project results indicate a practical understanding of testing methods
  - The interview results revealed decent theoretical know-how
  - Most students stayed motivated throughout the complete project
  - Most students properly handled their freedom of content choice and timing

- 
- Some tasks and phases consumed more time than expected
  - Balance of interviews and in-class support was not optimal
  - The ratio of actual group work was low
  - The scalability of the concept in its current form is limited

A+

- keep group size small
- limit dependency on group

- limit supervising effort
- keep student's timetables in mind

- adjust grading system
- make points more relative to effort

- generate and make use of a documentation

### References

[1] Laboratory project for a software quality class. <https://ix.cs.uoregon.edu/~michal/book/extras/Lugano-lab-2015/report.pdf>. Accessed: 2017-09-12.

[2] Franziska Böttcher and Felicitas Thiel. Der Fragebogen zur Erfassung studentischer Forschungskompetenzen. *Neues Handbuch Hochschullehre*, 11(2):57–64, 2016.

[3] Miriam Sophia Glessmer and Timo Lüth. Lernzieltaxonomische Klassifizierung und gezielte Gestaltung von Fragen. *Zeitschrift für Hochschulentwicklung*, 11(5):205–224, September 2016.