

# Software Lab 2019 - Guidelines

## Selection process

Every student that wants to participate must be registered for the course BV030004 in TUMOnline. After the announced time the students choose 8 topics on the main web page of the Chair for Computation in Engineering (<https://www.cie.bgu.tum.de/de/lehre/softwarelab>):

- 2 Gold topics
- 2 Silver topics
- 2 Bronze topics
- 2 No-go topics

In the end of the selection period groups of maximum 2-3 students will be formed. Bigger groups are not allowed, if not specified otherwise by the supervisors of the corresponding topic. Requests to have specific team members will not be considered when distributing the groups. The topics are distributed according to the most desired ones for a student. The “No-go” topics won’t be assigned.

## Tasks

- In the first arranged meeting the supervisors will explain the detailed content of the tasks together with the expected evaluation criteria. All necessary information, literature and project tools will be provided.

## Workflow

- Every meeting must be documented by the meeting protocol, which in the end of the course should be handed in to your supervisor. Not creating protocols on a regular basis will have a negative impact on your grade.
- A meeting protocol can be done using an issue tracker, on paper or with Excel.
- Regular meetings are highly encouraged (at least once per two weeks).
- Absence in the regular meetings without a valid excuse has a negative effect on the Team contribution and might result in a worse grade.
- Usage of a versioning system is mandatory. We suggest LRZ GitLab which is free for university students ([https://www.lrz.de/services/netzdienste/gitlab\\_en/](https://www.lrz.de/services/netzdienste/gitlab_en/)).
- Individual contributions will be valued in grading.
- Write clean code and document sufficiently (preferably using proper documentation tools such as DOXYGEN, Javadoc, ...) in English language.

## Evaluations

During the course there will be 3 main evaluations in the form of presentations. The last presentation is combined with a poster session. The students must hand in 3 presentations and the poster to their course organizer and their supervisors.

- Before each evaluation the deadline for handing in the presentations (and the poster for the third evaluation) will be announced by the course organizers. Violating the deadline or requesting changes in the slides afterwards will result in a grade penalty of 0.3.
- All students are highly encouraged to contribute to discussions during the evaluation by asking questions. This has a positive effect on the partial grade.
- A presentation will be interrupted if it extends the given timeframe.

- If a student does not attend a presentation without a valid excuse, their individual contribution to the presentation will be graded 5.0.
- Supervisors should be consulted before each presentation to practice the talk and review the slides. Same holds for the review of the poster before the last presentation.

### **Completion**

By the end of the course the students should hand in the completed code together with the requested documentation to their supervisors only. It is encouraged to use documentation tools like Doxygen to complete the task. The documentation must be written in English. Any violation of the deadlines will result in a lower grade.

## Software Lab 2019 – Evaluation criteria

### Final grade

Completing provided task will be graded 2.0. The supervisor will address possible advanced questions and tasks to improve the grade. The final grade will be composed of the following parts:

- Presentations (10% each)
- Teamwork (10%)
- Documentation + Poster (10%)
- Code (50%)

All the grades are given by all supervisors and discussed commonly.

### Penalties

- Each violation of a deadline degrades the respective grade by 0.3.
- In case of extreme delays, the associated category is graded with 5.0.
- Changes in the presentations, posters, code and documentations after the deadline count as deadline violations.

### Detailed description of each evaluation criterion

For the **presentations** 50% of the grade is given for the presentation itself and 50% for the current project progress.

The following questions will be taken into consideration for the evaluations:

- Were the contributions of all group members equally distributed?
- Was the presentation easy to follow?
- Was the presentation material complete and precise?
- Did the presentation seem well prepared?
- Did the students contribute to the discussions?
- Was the presentation finished in time?

The **teamwork** will be evaluated according to the following points:

- Did the students meet agreed milestones? (Time management)
- Did the team organize autonomously or did the supervisor have to assign tasks?
- Did the team try to solve problems autonomously or did they ask the supervisors immediately?
- Did the team meet with its supervisors regularly?
- Was the workload equally distributed between the team members?
- Did the students create protocols on a regular basis?

The **documentation** and the **poster** will be evaluated by the following criteria:

- Is the poster well structured?
- Does the poster motivate the topic?
- Does the poster motivate choices that have been made?
- Does the poster cite sources?
- Does the documentation serve to understand and use code?
- Does the documentation show the understanding of the task?

The **code** will be evaluated as follows:

- Does the code solve the assigned problems?

- Does the code run without compiler warnings, problems or errors?
- Was the code written completely by the students or with the extensive help of the supervisors?
- Was the code written in a clean and structured manner?

**A student does not pass if:**

- the final grade is worse than 4.0
- OR
- one category (individual presentations, report, poster, code and team contribution) is graded 5.0.

Unforeseen issues will be handled internally by the group's supervisors and the software lab organizers.