Software Engineering Lab

Organizational Matters

Prof. Dr. Sven Apel
Dr. Norman Peitek
Universität des Saarlandes

Based on material by Andreas Zeller and Norbert Siegmund

SE Lab Team















Learning Goals

- Acquire the ability to **solve complex software development problems** individually and in teams.
- Gain awareness of **problems and pitfalls of software development** and know how to address them.
- Accomplish and **coordinate** software development tasks based on a set of given requirements.
  - For this purpose, you are able to select proper methods and techniques to minimize risks and maximize software quality.
- Learn foundations and **principles of software design**, including cohesion, coupling, modularity, encapsulation, abstraction, and information hiding.
  - Get acquainted with a whole array of **design patterns**, knowing their aim and individual strengths and weaknesses.
  - Able to apply design patterns beneficially and to judge and improve the quality of software designs.
- Master fundamental techniques and tools for **software testing**, debugging, and version control.

Theoretical Lectures	Practical Lectures	Exercises	Group Project

SE Lab Evolution (since 2022)

2022

- Quality of life improvements
- Practical tutorials

2023

- Kotlin
- Deterministic projects

2024

- More transparency in passing criteria
- Extended test descriptions

2025

- No individual phase
- More time for group phase
- Reduced compulsory attendance

CMS Web Page

https://cms.sic.saarland/se_lab_25/

Software Engineering Lab (Software-Praktikum)

Prof. Dr. Sven Apel, Dr. Norman Peitek

Summer Semester 2025

Registration for this course is open until **Monday, 25.08.2025 12:00**.

News

Currently, no news are available

In case of questions, please contact us via e-mail to: sopra@lists.se.cs.uni-saarland.de.

About the Software Engineering Lab

Passing the course "Programming 2" is a mandatory prerequisite for participation in the Software Engineering Lab!

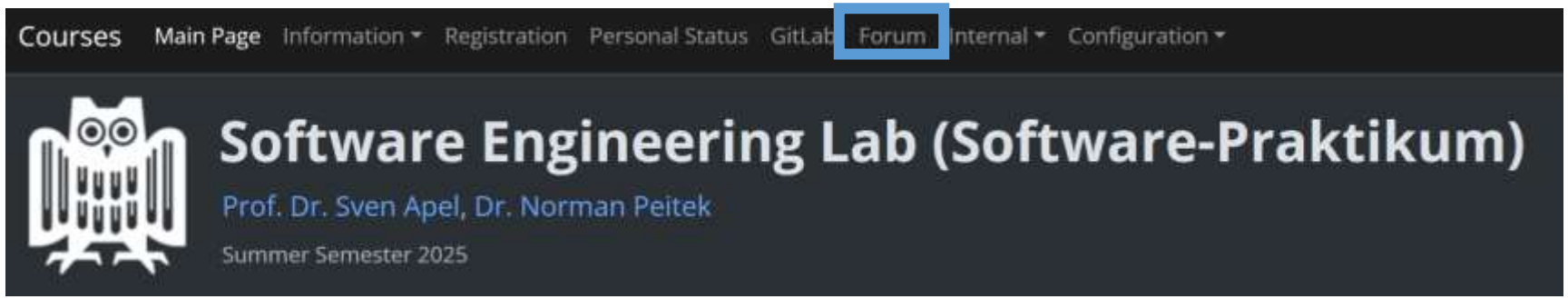
The Chair of Software Engineering offers a Software Engineering Lab (SE Lab), formerly Software-Praktikum (SoPra), in the form of a basic block course during the lecture-free period, which addresses students currently enrolled in a **Bachelor's degree program** majoring and minoring in computer science. The goal of the Software Engineering Lab is to develop a non-trivial software system in Kotlin in a group effort. Other characteristics of the Software Engineering Lab are:

- Time period: 25.08.2025 – 10.10.2025
- Duration: 7 weeks (daily Monday to Friday)

CMS Registration:
by today at noon!

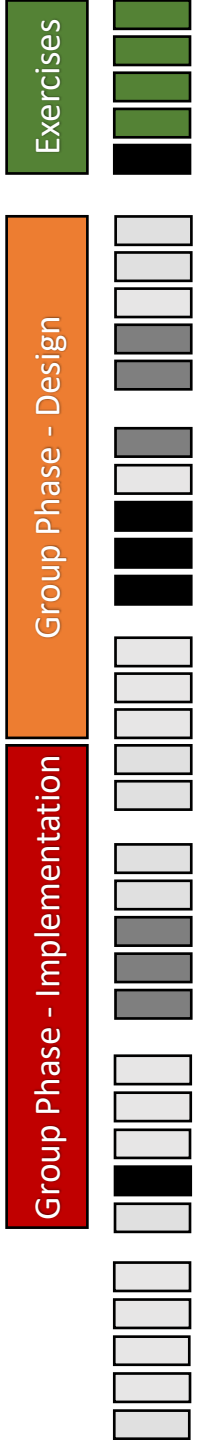
Discussion Platform

https://forum.se.cs.uni-saarland.de:51443



Schedule

~6 weeks full-time!



Schedule

Exercises

Start of Exercise Phase 25.08.

Entry Exam 29.08.

Group Phase - Design

Practical Tutorial 01.09
Start of Group Phase 02.09.

Design Review
04.09. – 08.09.

Design Defense
10.09. – 12.09.

Design Re-Evaluation
15.09. – 17.09.

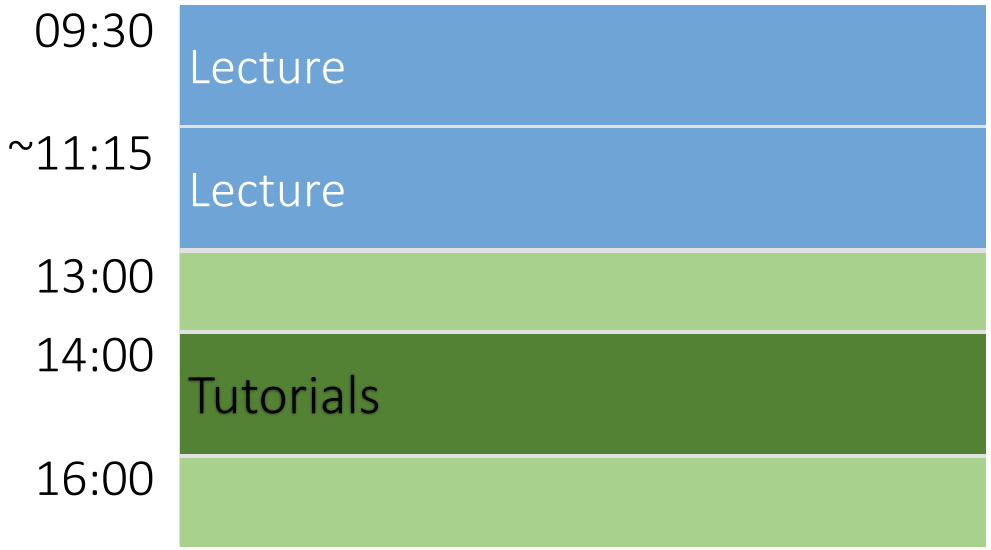
Group Phase - Implementation

Code Review
24.09. – 26.10.

Code Submission 02.10.

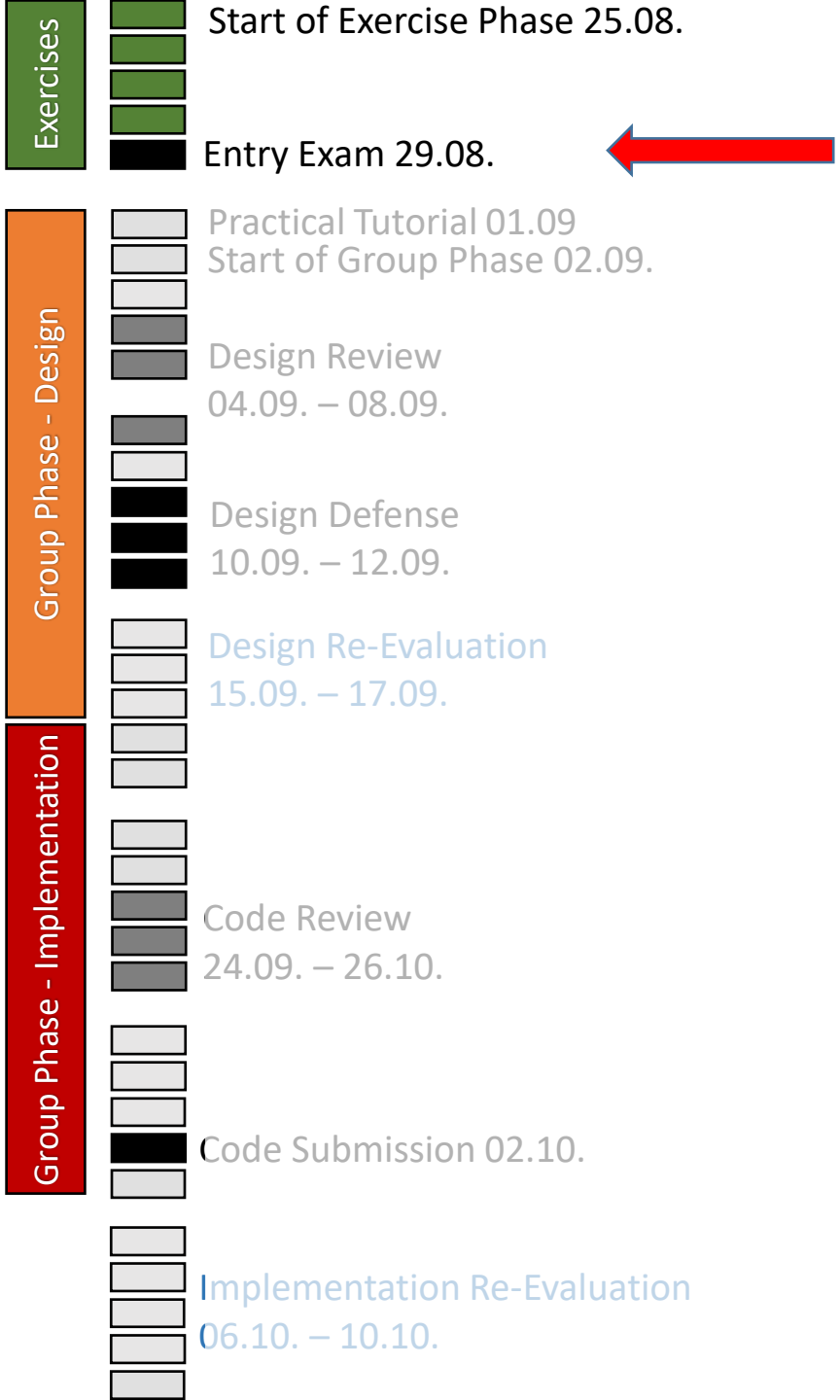
Implementation Re-Evaluation
06.10. – 10.10.

Exercise Phase



- Exercise sheets in the tutorials
- Special mock exam exercises to prepare for the entry exam

Schedule



Exercise Phase

Entry Exam (29. August)

- Be there at 09:45!
- 60 Minutes
- Topics of lectures, tutorials, and exercises
- Passing the entry exam is admission requirement for the group phase
- Bring student **and** photo ID

LSF Registration: by Sep 01st!

Schedule

Exercises

Start of Exercise Phase 25.08.

Entry Exam 29.08.

Group Phase - Design

Practical Tutorial 01.09
Start of Group Phase 02.09.

Design Review
04.09. – 08.09.

Design Defense
10.09. – 12.09.

Design Re-Evaluation
15.09. – 17.09.

Group Phase - Implementation

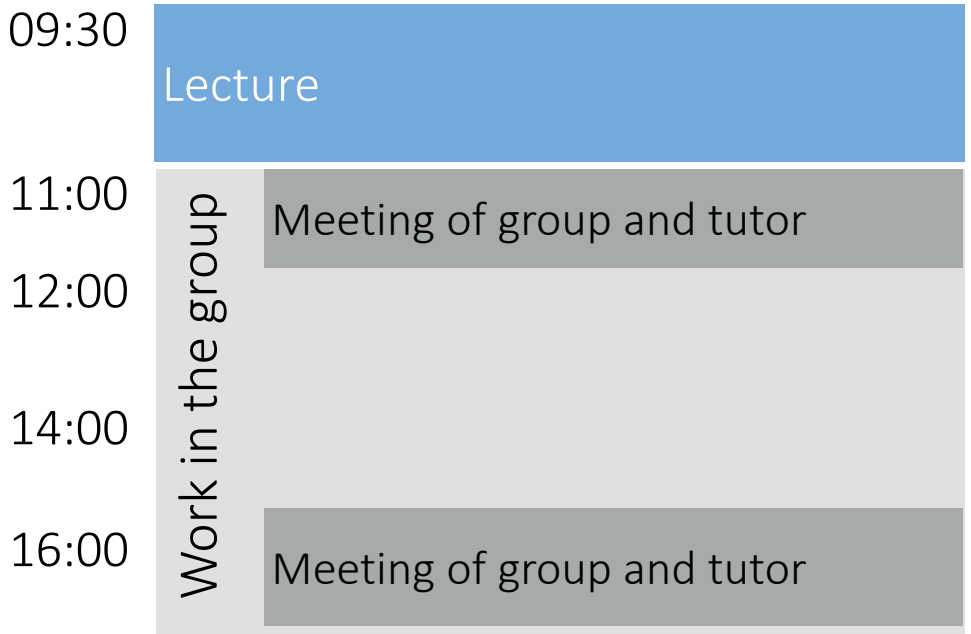
Code Review
24.09. – 26.10.

Code Submission 02.10.

Implementation Re-Evaluation
06.10. – 10.10.

Group Phase

At days with a lecture:



Practical Lectures & Tutorials

- Practical lecture *Advanced git* and *Teamwork*

- Lectures on *Code Review* and *Continuous Integration*

- Practical tutorial on *Testing and Debugging*

- Guest lecture by Raphael Nömmer
  - Topic: *Testing in Practice*

CQSE

Schedule

Exercises

Start of Exercise Phase 25.08.

Entry Exam 29.08.

Group Phase - Design

Practical Tutorial 01.09
Start of Group Phase 02.09.

Design Review
04.09. – 08.09.

Design Defense
10.09. – 12.09.

Design Re-Evaluation
15.09. – 17.09.

Group Phase - Implementation

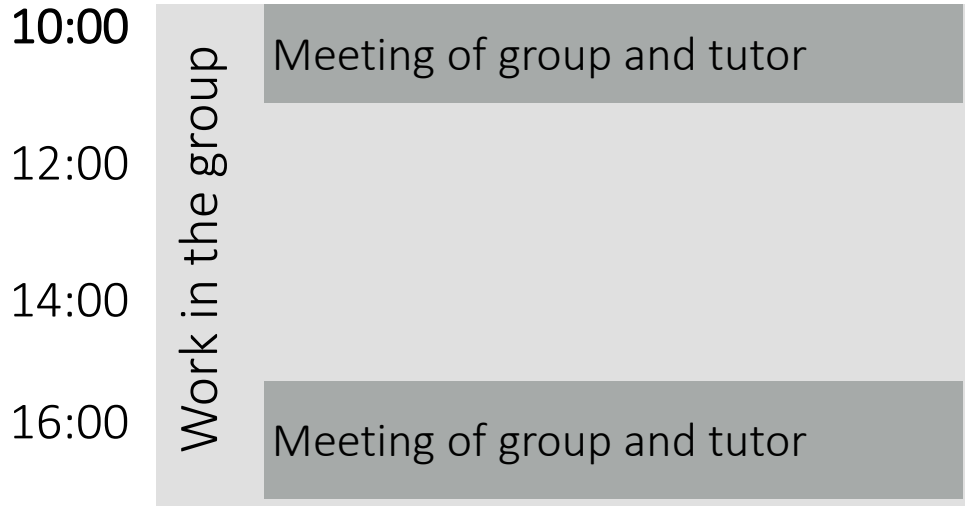
Code Review
24.09. – 26.10.

Code Submission 02.10.

Implementation Re-Evaluation
06.10. – 10.10.

Group Phase

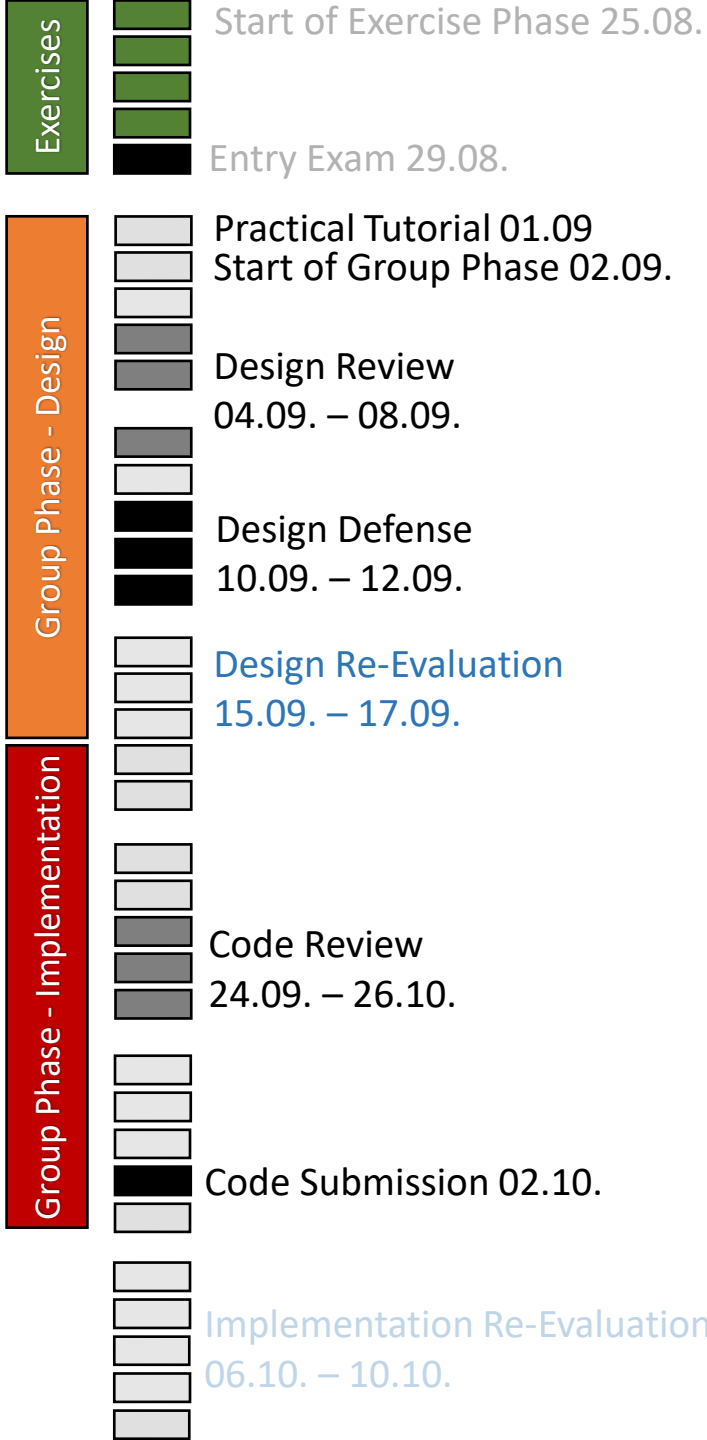
At days without a lecture:



New: Attendance is not required for one day of the week
(Default: Tuesday, but group decision)

Schedule

Group Phase



Tasks

- Development of a rough and then detailed software design
- Implementation & Testing

Design

Exercises

Start of Exercise Phase 25.08.

Entry Exam 29.08.

Group Phase - Design

Practical Tutorial 01.09
Start of Group Phase 02.09.

Design Review
04.09. – 08.09.

Design Defense
10.09. – 12.09.

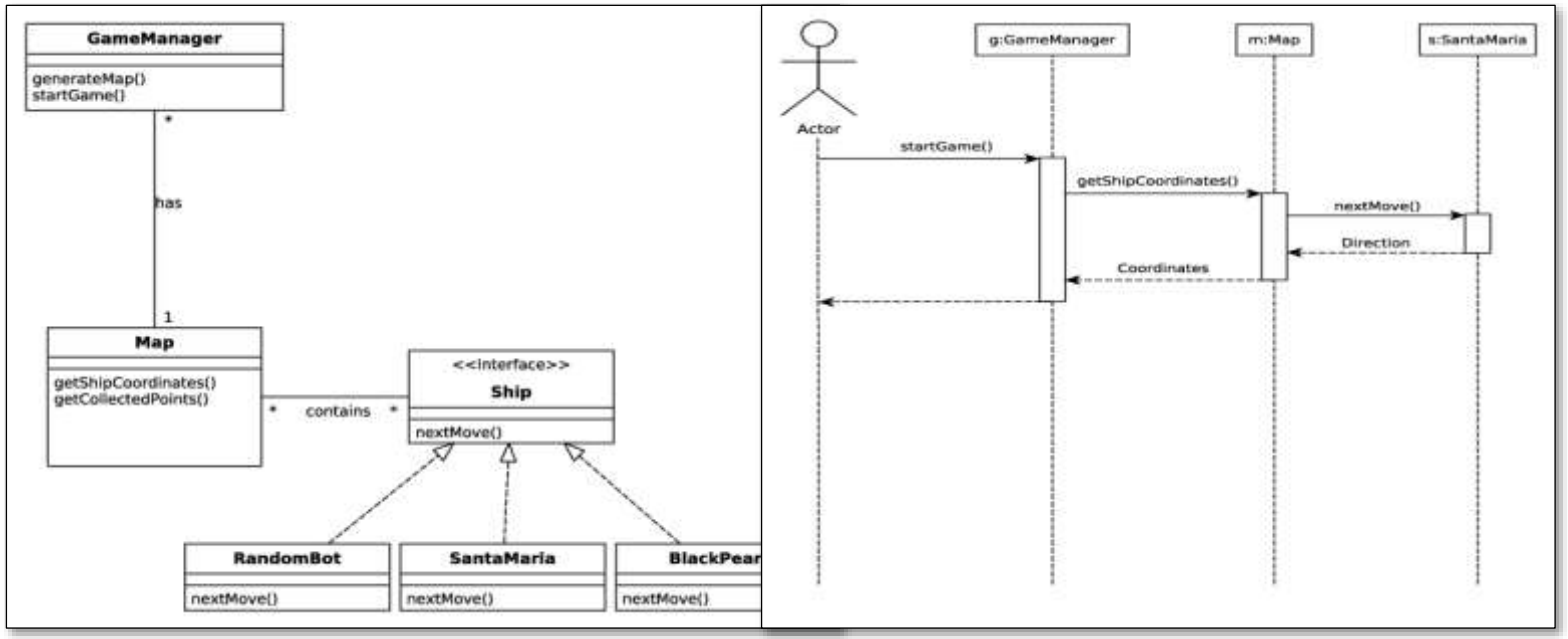
Design Re-Evaluation
15.09. – 17.09.

Group Phase - Implementation

Code Review
24.09. – 26.10.

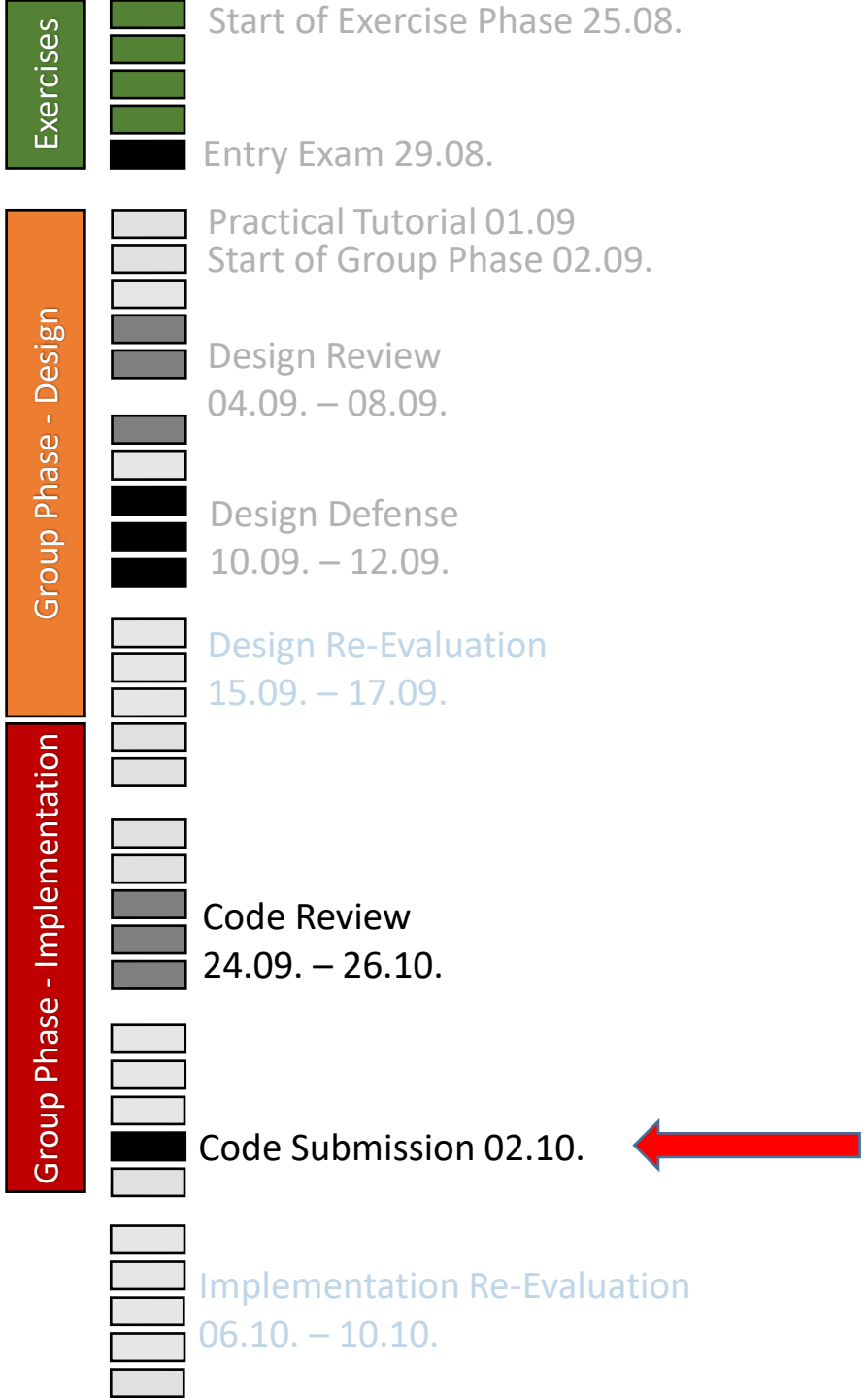
Code Submission 02.10.

Implementation Re-Evaluation
06.10. – 10.10.



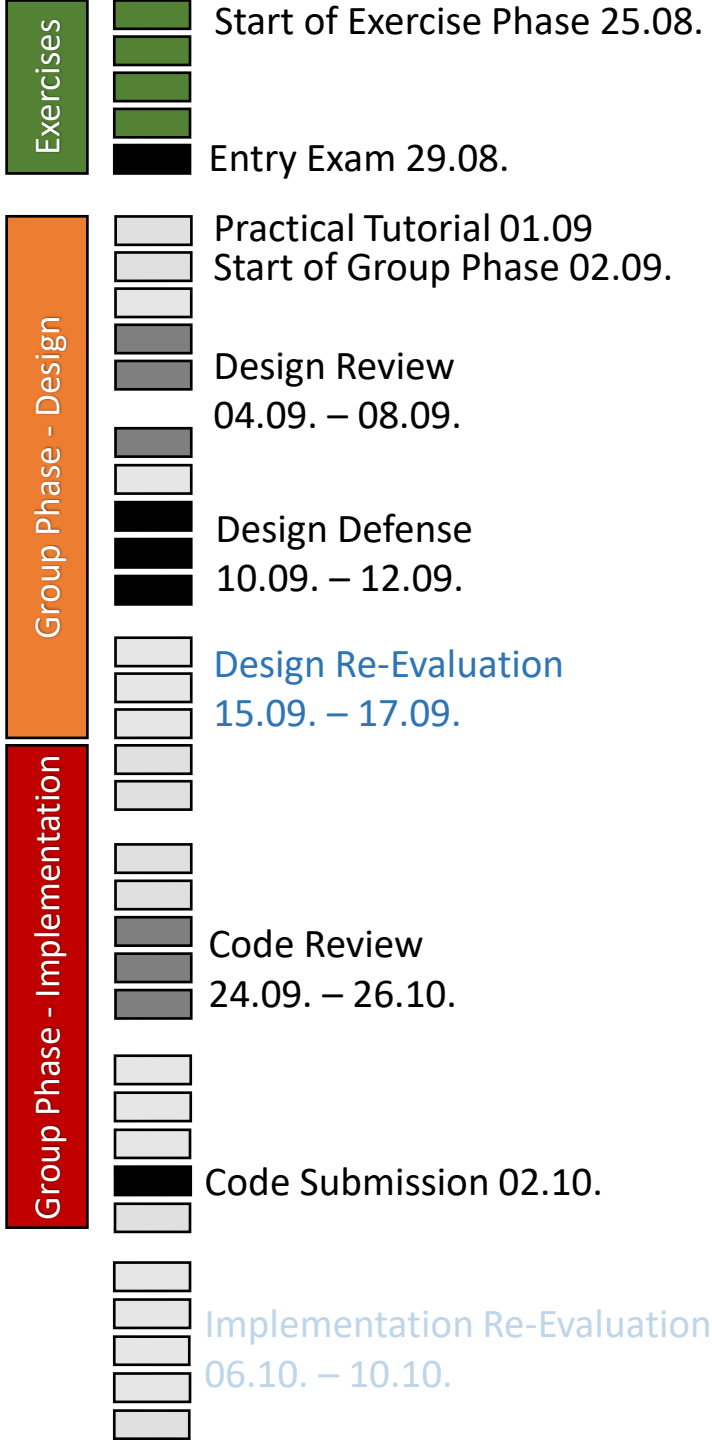
Design Defense

- If you pass the design defense, you get a break
- If you do not pass the design defense, you have a second chance in the design re-evaluation



Implementation & Testing

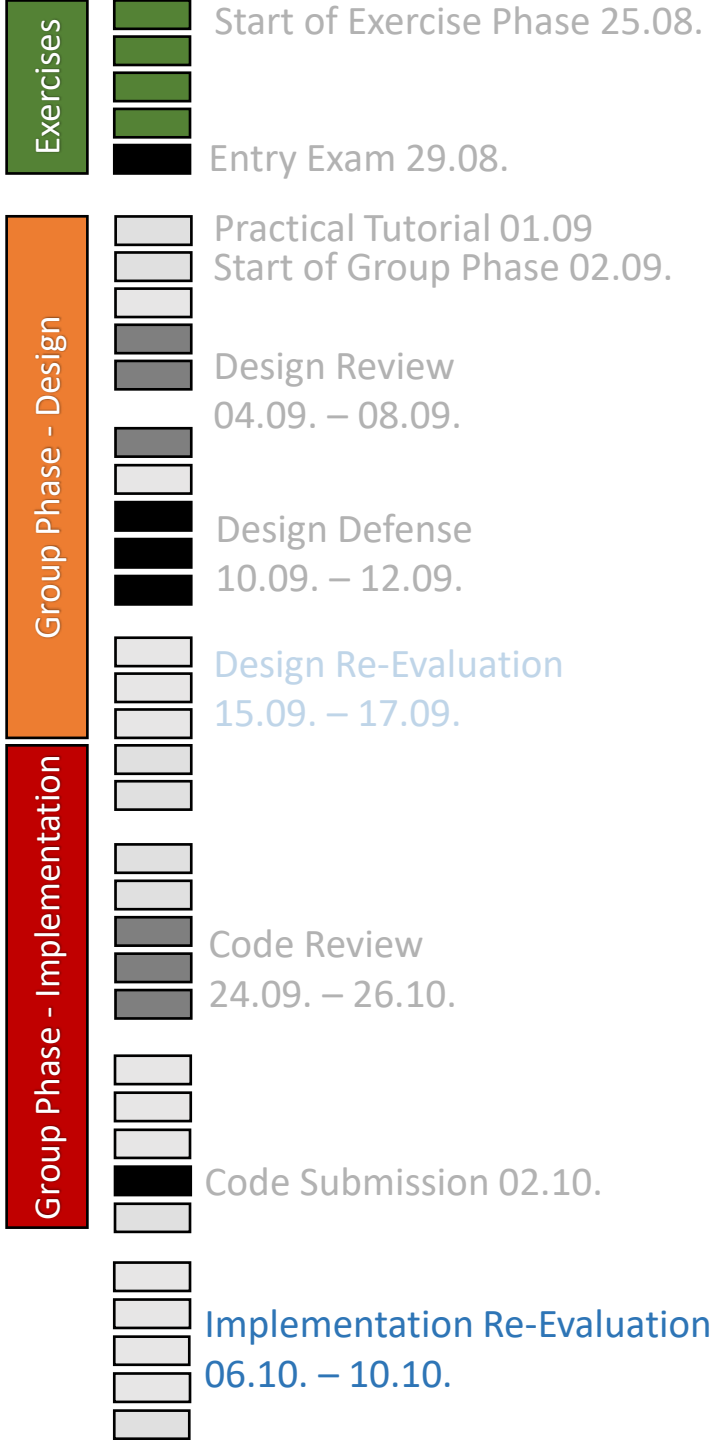
- Pass *our* tests
  (public, daily tests)
- Pass *your* tests
  (and find mutants in our/your code)
- High-quality, comprehensive tests
  (in line with the concepts from the lectures)
- Pass code-analysis tools
  (e.g., Detekt)
- Code and tests must be well-structured
  (according to the concepts taught in the lecture)
- Every single group member has contributed to the code sufficiently
  (checked via project report and git commits)



Passing the SE Lab

Passing the course "Programming 2" is a mandatory prerequisite for participation in the Software Engineering Lab!

1. You are registered in the CMS and for the exam (also in the LSF, exam registration and withdrawal are possible until 01.09.).
2. You pass the entry exam.
3. Your design fulfills the principles of object-oriented design.
4. Your implementation passes all tests.
5. Your tests identify mutants.
6. Your code and tests are appropriate and of high quality.
7. You individually have clearly communicated your contributions that significantly to the success of your project in the implementation plan, which is also verifiably (git commits)

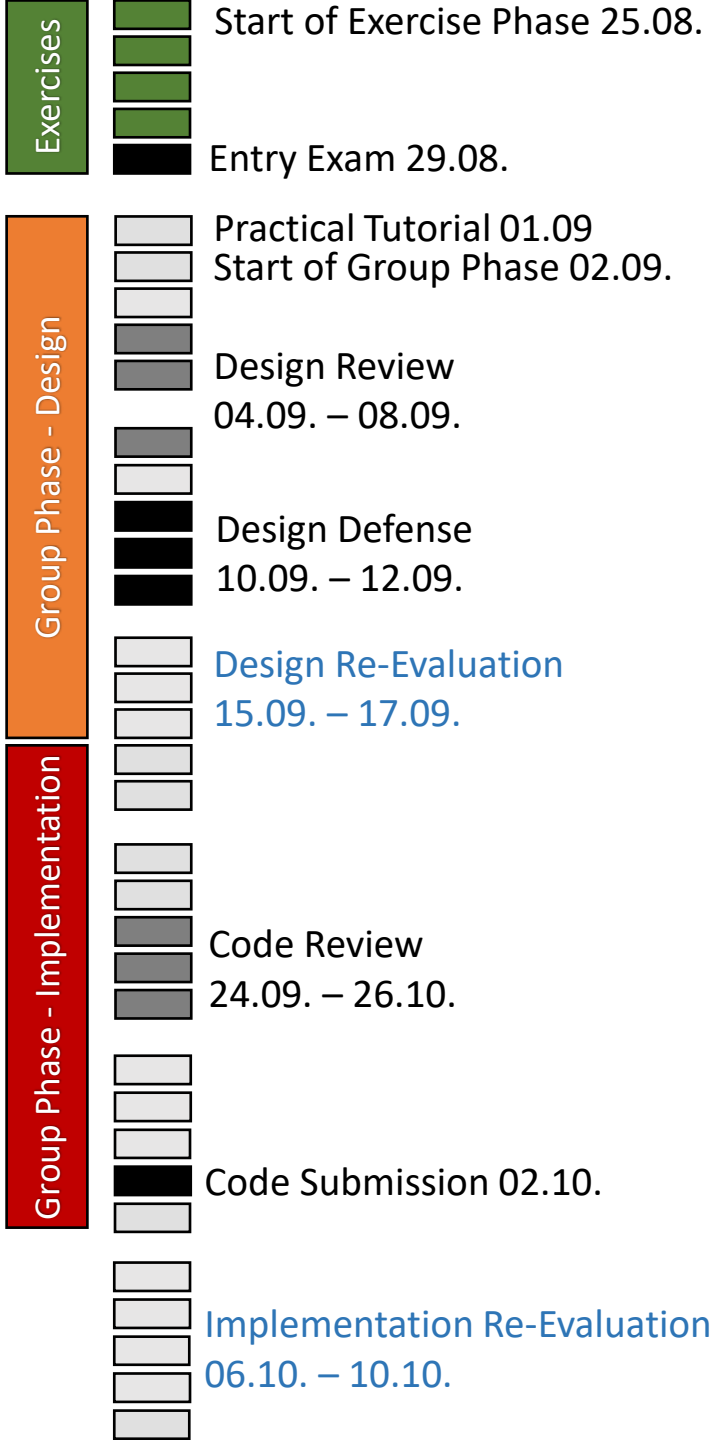


Re-Evaluation

- If you have passed the group phase, you have completed SE Lab
- If you or your entire team have **not passed** the criteria for the group phase, you can go into the re-evaluation week
  - Here, we will conduct one-on-one oral exams, in which you can defend your individual contributions to group project

Course Language

- **Lectures and materials:** *English*
- **Entry exam:** Tasks are formulated in *English* and *German*.
  - You can provide your answers in either of both languages.
- **Tutorials in the exercise phase:** *English* or *German*
  - (choosing the preferred language was possible until yesterday)
- **Project work in the group phase:** *English* or *German*
- We try to respect your preferences, if possible, but without guarantee.
  - *Default language* is *English*.



Questions?

Email: sopra@lists.se.cs.uni-saarland.de