Lecture "Software Engineering SoSe 2023

Free University of Berlin, Institute of Computer Science, Software Engineering Group Lutz Prechelt, Linus Ververs, Oskar Besler, Tom-Hendrik Lübke, Nina Matthias

Exercise sheet 1 **Introduction** to 2022-04-24

Work on all tasks in such a way that you are **well prepared for** the exercises, i.e. that you can **show**, **explain** and **discuss** your solutions in the exercise group in an appropriate way. Always indicate your **sources** used.

Task 1-1: Organizational, KVV

a) Make sure that you are registered in the **KVV** for the course "Softwaretechnik S22". Register in a **tutorial** (via the menu item *Section Info*).

Note: The tutorials do not start until the second week of the lecture, this exercise sheet has to be worked on before.

b) **Subscribe to the forum of the** event **in the KVV** (via the *Watch* menu item) and skim any existing threads so that you do not miss any relevant information. During the semester, the forum will serve as the communica- tion platform for the event. If you have any questions about the course, lecture content or individual assignments, please post them there.

c) **Read** the Active Participation criteria in the "Announcements" in the KVV.

d) Please work on the exercise sheets **in pairs**. Remember to **upload** your solutions always before Monday 12:00 (s.t.) via KVV and please **write both names** on the submission; one upload per exercise partner group is sufficient**.**

If you do not yet have a tutorial partner, work on the first exercise sheet alone; we will then find someone for you in the first tutorial. In exceptional cases, cross-tutorial pairs are okay if you have the same tutor and he/she agrees.

Task 1-2: Conflicting goals in software development

A software development is generally considered successful if

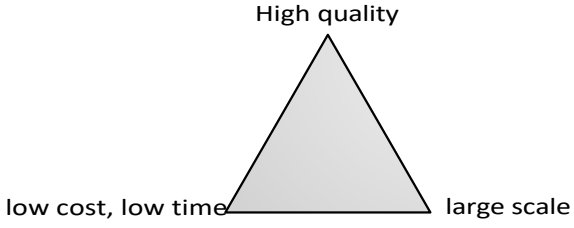
- the **costs** were low,
- the elapsed **time** to completion was short,
- the **quality of** the software is high and
- the **scope of** the software is large, so many functions have been

implemented. Work on the following subtasks:

a) Are all four goals equally achievable? What happens if you try to optimize one of the goals (e.g. quality)?

b) Give and explain one example of each type of software where one of the four goals should take precedence.

c) Why is it possible to combine the goals of low cost and low time expenditure in software projects and thus represent the goals of software development as a **triangle?**



d) Examine the relationship between **cost** and **time in** a software project in more detail.

Assume that a software developer would need 1 year development time for a software with defined scope and quality. How long would two, five and one hundred and fifty developers need for the same project? Give reasons for your answers.

Task 1-3: Clarification of terms

The following terms are part of the basic vocabulary of (object-oriented) programming and should already be familiar to you, e.g. from previous courses in computer science. We will use them routinely during the software engineering course, so it is important that you develop a clean understanding. Refer to appropriate sources if you are unclear, and cite them as always.

a) For each term, formulate a concise explanation in your own words and illustrate it with an example of your own choosing. If there is a relationship between two terms, explain it as well.

> *Class, Inheritance, Attribute, Operation, Method, Copy, Object, Library, Specification, Implementation, Verification*

b) Explain the difference between a *class* and a *set of objects*.

c) Which software engineering ac- tivities mentioned in the first lecture of 2022-04-21 are *not* covered by the terms in **a) above**? Give two examples that are as clear as possible.