Lecture "Software Engineering" summer semester 2023

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

exercise sheet 2   **Own idea for a software project**   to 2022-05-01

Voluntary compulsion: There are 2 ways you can take part in the software engineering tutorial:

1. Regular practice sessions - as you know it from other events
   - with the**obligatory, weekly submission of exercise sheets and participation in the tutorials.**At the end of the semester, a lower limit of points must be reached across all exercise sheets (the exact point limit can be found in the announcements on the whiteboard).

2. Alternatively, you can decide to be exempted from the obligation to submit the exercise slips. Instead, you will receive one**particularly active participation in the tutorials**(through clever questions and verbal contributions, active participation in group work, short presentations on related topics, calculations, etc.). Your tutor will give you feedback on your status during the semester and point out if the existence of active participation is at risk due to insufficient participation.

*A notice:*Option 2 is aimed at intrinsically motivated students who want to deal with the lecture topics (and more) without being forced to submit an exercise sheet. Choosing the 2nd option will not lead to time savings, since you have to prepare for the tutorials in order to actively participate and also have to master the material on the exercise sheets for the exam. With option 2 you also miss the feedback from the exercise sheet correction. However, you can only submit an exercise sheet for correction twice a semester, for which you urgently want feedback.

Please use this exercise sheet to decide which option you would like to choose for the rest of the semester and mark this decision as part of this submission. If you choose option 2, you do not have to work on the remaining tasks.**The choice you make here is binding for the entire semester.**

Edit all tasks so that you can use them for the exercises**well prepared**are, ie that your solutions in the exercise group are in a suitable form**show**,**explain**and **argue**can. Always give your used **Sources**at.

Task 2-1: Software engineering tasks

Learning objective: Knowing and understanding the tasks and areas of software engineering.

a)Recall the*Tasks*of software engineering:

    1.Which are they?
    2.What are they each dealing with?

b)Assign the tasks to the respective*areas*of software engineering.

    Areas of software engineering:*Quality Assurance, Design/Implementation, Requirements, Management*

Task 2-2: Develop your own idea for a software project

On the following exercise sheets, the different areas of software development and their activities (such as the*requirements determination*,*analysis*and *modelling*) can be made tangible using a concrete example.

Develop your own **realistic idea** for software to be developed. You should use this software as part of the event **not implement**.

- Your idea should be a relatively simple software that is not too complex. For example, it should not be a computer game.
- The software should interest you thematically and/or you should see a professional or private (e.g. hobby, association) purpose.
- This can be an idea that you have already dealt with in some way, but it can be **not** be a finished software project.

Complete the following subtasks:

a) Describe your idea for software to be developed in text form. Please consider the following aspects:

- What is the application domain of the software?
  What kind of software is it, what is it needed for?
  - Which work process should it support/improve?
  - What functions should it provide?
  - What is the relationship between the costs and the benefits of the software?
- Who are your customers (ie where does the – imagined – order and the money for the development come from)?
- Who is the target group of your software?
  What level of knowledge is required or expected from the users when using the software?

  What problems / resistance to the introduction of the software can be expected?

b) Now prepare the description of your software idea as a Pecha Kucha presentation (see below) so that you can present it to your fellow students in the exercise.

- Export your presentation as **PDF**!
- **Submit your presentation together with your exercise sheet solutions under Assignments** and link your slides (e.g. via FU-Box **box.fuberlin.de**) on their wiki page (see below).
- The file name of your presentation should conform to the format   bad luck Ku cha_[Last Name1]_[Last Name2].pdf   are equivalent to.

c) There will be several tasks over the course of the semester that build on this software idea. Therefore set up one in the KVV **wiki page** on which you summarize all partial results.

- The name of the wiki page should conform to the following scheme:
  [Per project n/Ame]: [After n/Ame1], [After n/Ame2]
- On the wiki page itself, please be sure to include the following information as well: **Project name, tutor, full names of both participants**.
- Make the description out task **a)** also on the wiki page. this applies **additionally** for the normal, electronic submission of your solutions.
- Link the presentation from assignment **b)** also on your wiki page.

For the upcoming exercise sheets: Make sure that the wiki page always clearly identifies which part goes with which exercise.

PechaKucha(*[petscha-kutscha]*, "chatter," "chit-chat") is a presentation method. It is based on the knowledge that the listener's attention wanes after about seven minutes. That's why PechaKucha lectures are structured according to strict rules:

- The oral presentation is accompanied by exactly 20 slides or images.
- The speaker has exactly 20 seconds for each slide, resulting in a fixed presentation duration of 6 minutes 40 seconds.
- The slides change automatically. (Your tutor will take care of this.)

These guidelines force you to present the topic in a precise and structured way. Sources:

- http://pechakucha.de/berlin/
- http://pecha-kucha.org/night/berlin/
- http://de.wikipedia.org/wiki/Pecha_Kucha
- http://en.wikipedia.org/wiki/PechaKucha
- http://www.pressebox.de/pressemeldung/jordanize/boxid/268080