

Written exam for Software Engineering Theory

Course codes TDDC88, TDDC93, 725G64

Instructions to students, please read carefully

- **Explicitly forbidden aids:** Textbooks, machine-written pages, photocopied pages, pages of different format than A4, electronic equipment.
- Try to solve as many problems as possible.
- Motivate all solutions.
- Please, write and draw clearly.
- Write solutions for different areas (fundamental part) and different problems (advanced part) on separate sheets of paper.
- Label all papers with AID-number, date of examination, course code, examination code, and page number.
- You may write solutions in either Swedish or English.
- Please, note that the problems are not necessarily written in order of difficulty.
- **TIP!** Read through all exercises in the beginning of the exam. This will give you the possibility to ask questions about all parts of the exam, since the examiner will visit you in the beginning of the exam time. Remember that the examiner has over 200 students, so plan for solving problems in a flexible order if you have to wait for clarification.

Grading

The exam consists of two parts: Fundamental and Advanced.

The Fundamental part has problems worth 10 credits per area. Areas are: Requirements, Planning & Processes, Design & Architecture, Testing & SCM, and Software Quality. Thus the Fundamental part can give maximally 50 credits.

The Advanced part has problems worth 50 credits in total. Each problem typically requires a longer solution of several pages.

The maximum number of credits assigned to each problem is given within parentheses at the end of the last paragraph of the problem.

Pass condition: At least 4 credits per area in the Fundamental part **and** at least 50 credits in total. The total amount of credits also includes the bonus credits you might have got in lecture exercises autumn 2011. This gives you the mark 3 in the Swedish system and a C in ECTS. If you have at least 4 credits for 4 of the areas in the Fundamental part, then you can still pass if you have more than 60 credits in total.

Higher marks are given based on fulfilled *pass condition* **and** higher amounts of credits according to the following table:

Total credits	Mark in Swedish system	Translation to ECTS
0-49	UK	Fx
50-66	3	C
67-83	4	B
84-	5	A

Multiple choice questions

In multiple choice questions we will ask you to write down the letters A, B, C, or D for the one or two statements that you think are true. Note that you should not write down the statements that you think are false. There are exactly two true statements per question, so answering with three or four alternatives with gives 0 credits.

For each statement that you select that is correct (i.e., that the statement is in fact true) you get one credit. For each statement that you select that is incorrect (i.e., that the statement is in fact false, but you believed it was true) you get minus one credit. Each multiple choice question can give maximum 2 credits and minimum 0 credits, i.e., you cannot get negative credits for one multiple choice question.

Example 1: Assume that you have written down statements A and C. If now statements A and B were true, and statements C and D were false, you would get +1 credit for writing down A, but -1 credit for writing down C. Hence, the total credits for the multiple choice question is 0.

Example 2: Assume that you have written down statement B. If now statement A and B were true, and statement and statement C and D were false, you would get +1 credit for the multiple choice question.

Example 3: Assume you correctly wrote both statement A and B. If now statement A and B were true, and statement and statement C and D were false, you would get +1 credit for writing down A, and +1 for writing down B. Hence, the total credits for the multiple choice question is 2.

Good Luck!

Kristian

Problems

Part 1: Fundamental

Area 1: Requirements

1 a) Which of the following statements are true? Answer with the statement letter only, no motivation is needed. (2)

- A. The requirement “The system shall respond to the end-user within 3 seconds” is a functional requirement.
- B. The requirement “The project team shall use SCRUM with a sprint every 30 days” is a functional requirement.
- C. The requirement “The sum of the costs entered by the user shall be visible in the bottom row of the sheet” is a functional requirement.
- D. The requirement “The system shall be able to generate reports in PDF, Word, and HTML” is an example of a feature that can be further broken down to more detailed requirements.

1 b) Draw a UML Use-Case Diagram of a system for a web-based train ticket selling system. There shall be two different use-cases and two actors in the diagram. Use-case texts for the use-cases shall be written in complete sentences. (4)

1 c) Describe the concepts Requirements Elicitation, Requirements Analysis, Stakeholder, and Requirements Validation. (4)

Area 2: Planning and Processes

2 a) Which of the following statements are true? Answer with the statement letter only, no motivation is needed. (2)

- A. The risk “There might be a new law of automated trading” is an Indirect Risk for a project developing an automated trading system.
- B. The Risk Magnitude Indicator is the Risk Probability multiplied with the Risk Likelihood.
- C. Risk Mitigation means lowering the impact of a risk if it comes true.
- D. Risk Monitoring means that we reassess the Risk Probabilities on a regular basis during the project.

2 b) Describe two disadvantages with the traditional Waterfall Life-Cycle Model. For each of these disadvantages, explain how a project using an Agile Method can avoid them. (4)

2 c) Name and describe the four Dependent Project Parameters. (4)

Area 3: Design and Architecture

3 a) Which of the following statements are true? Answer with the statement letter only, no motivation is needed. (2)

- A. The Implementation View of an architecture can help a programmer finding the source code of a certain function.
- B. The Deployment View of an architecture shows which software packages that are installed and run on different machines.
- C. The Structural Views of an architecture show how changes between states of the system will occur in real time.
- D. The Rationale of an architecture description document gives instructions of how to implement the source code in a rational way.

3 b) Describe two advantages and two disadvantages with a Layered Architectural Style. (4)

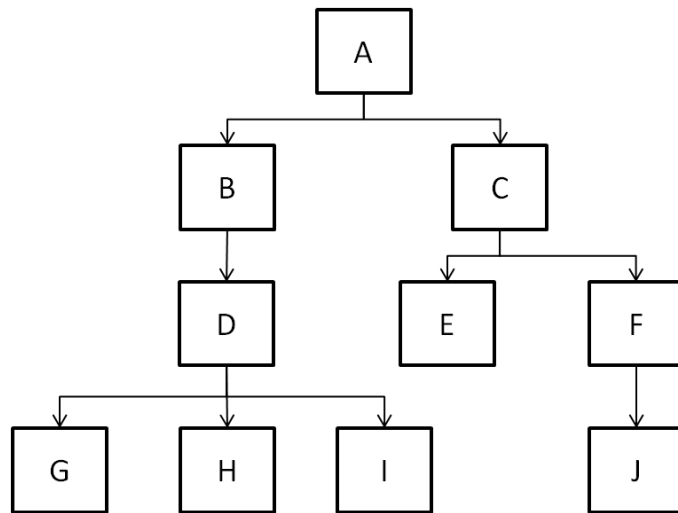
3 c) Describe the following concepts of a UML Sequence Diagram: Message, Life Line, Alt Fragment (aka nested conditional), and Guard Condition.(4).

Area 4: Testing and SCM

4 a) Which of the following statements are true? Answer with the statement letter only, no motivation is needed. (2)

- A. When you are ready with your code you make an Update in Subversion to send your changes to the repository.
- B. Git is a revision control system that supports distributed development.
- C. The Trunk of a code repository contains the “main line” of development.
- D. A major draw-back of the old Lock-Modify-Unlock revision control system is that it only works for two developers.

4 b) Below you find a decomposition tree of an imaginary system of 10 modules. The arrows indicate dependency between modules. Describe the order of integration and testing if you are using a Bottom-Up integration strategy. Give one advantage and one disadvantage of Bottom-Up integration testing. (4)



4 c) Describe the following terms in the context of Unit Testing: Oracle, Equivalence Testing, JUnit, and Test-Driven Development. (4)

Area 5: Software Quality

5a) Which of the following statements are true? Answer with the statement letter only, no motivation is needed. (2)

- A. The Acronym PDCA as used in the Shewhart Cycle means Program, Deploy, Compile, and Analyze.
- B. Total Quality Management means that all quality factors are equally important, and must be treated in a similar way.
- C. In ISO 9001 it is important not to press suppliers too hard to get the lowest prices. They need to keep some room for innovation to the benefit of both suppliers and buyers.
- D. TickIT is a certification organization with an interpretation of ISO 9001 for software.

5 b) Suppose that you work with an organization that is mature when it comes to understanding customer requirements, and there are few problems with developing the first version of the system. However, the work with subsequent versions and maintaining good quality is problematic. Suggest 4 different CMMI Process Areas that you think are the most important for the organization. Write a short motivation for why you chose each of the Process Areas. (4)

5 c) Describe the following concepts of Software Inspections: Inspection leader (aka Moderator), Inspector, Entry Criteria, and Exit Decision. (4)

Part 2: Advanced

6. Suppose you are the manager of a company developing mobile apps for iPhone and Android phones. Your market niche is personal health management, for instance, nutrition advice, step counter, training schedule, etc. From the nature of your business you have derived a quality vision for your present and coming products: Your software must be attractive to get selected by the users in the first place. The software must be easy to work with, but at the same time support long-time usage, including migration to new versions of the platforms and new types of devices, such as iPads. The correctness and reliability are important since it is about the health of the users, but it is not as life-critical as software in hospital care. You have succeeded in creating an innovative climate amongst your employees, pricing is not a problem. However, since you are growing both in number employees and customer base, you wish to be clearer on quality.

The text above is more like a vision which is hard to evaluate. Your task is to formulate five concrete and quantitative SMART¹ goals for the product quality of your company. By quantitative we mean that there shall be a single number in the goal text, which can be changed depending on ambition level. For instance, if your goal is to attract 1000 customers, you can increase the ambition level by striving for 2000 customers instead. For each of the goals, you shall provide an argument for why your goal is a SMART goal.

You shall also give detailed descriptions of what you will measure and how you calculate the numbers that can be compared to your goals. Relate each of the goals to one or more Software Quality Factors that we talked about in the course.

Don't forget to motivate your choices so that it is clear that there is a connection between the goals and the vision. You may make more assumptions about your products as long as they are clearly written down. (20)

¹ Forgotten what SMART means? Specific, Measurable, Agreed upon, Realistic, Timely

7. You have got an order for a University Scheduling System. Inputs to the system are:

- The classrooms and lecture halls with size and available equipment.
- Equipment possible to rent for a room.
- The courses with expected number of participants, number and size of lecture and lab groups, and the time slots of the schedule blocks to which the courses are allotted.
- The wishes from the teachers of times suitable to book for different activities.
- Hard constraints from teachers, for instance, if there are necessary travels.
- Wishes from the student union as regards exams, for instance, exams that cause many collisions or uneven distribution of exams.

Output is a schedule of bookings, where each booking contains information about course codes, type of activity, student group identification, assigned room, assigned teacher, and booked extra equipment. Many different reports can be generated for students, teachers, room cleaning personnel, exam vigilators etc. The schedule database can be searched with structured queries or free text, and it is possible to synchronize, the result of a report or search with popular calendar programs.

The output is trying to fulfil all wishes and constraints. If this is not possible, the system warns the administrator who makes bookings manually to make a good compromise.

The quality of information integrity is very high, since a faulty schedule will generate much cost and frustration.

Now it is time for you to design the system.

a) Create a UML Class diagram to describe the information elements handled by the system. The reason for selecting the classes and attributes shall be written down. The relations between the classes and their multiplicities are important. (10)

b) Draw an execution view block-and-arrow diagram of two different architectures of the scheduling system. Describe the advantages and disadvantages of each of the architectures. Which one do you think is the best for the system? Motivate your answer. (10)

c) Write a suite of at least 5 test-cases that can be used for automatic smoke tests on system level of the schedule for a single course. Describe how the test-cases can be executed automatically. To do this you must make assumptions of the input and the test database. For instance: The course has 100 students, one teacher and a written exam. The teacher cannot take any lectures before 10.00, and is absent during week 4 of the course. At most one lecture can be scheduled in a single day. The students want the exam in the second half of the examination period, but not on a Saturday afternoon. If that happens, they can accept examination in the first half of the examination period, as long as it isn't given the same day as the nightmare course TDDD666. Write down your assumptions. (10)