T-76.4602 Software Development Methods
Examination, 16th December, 2009

Write the following information in each paper you return: name, student number, course code and name, date, signature. Answer all questions of Part A. Part B of the exam you can take with you and return it by the 26th of January.

Part A. There are four questions (1, 2, 3 and 4), each worth 6 points.

1. Requirements engineering. Answer a, b, and c.

a) Describe in short which things are represented/modeled in a use case diagram and how use case diagrams and use case descriptions relate to each other?

b) Describe in short what customer-centric orientation (cultural change from a technology-centric view to a customer-centric view) means from the perspective of software engineers and IT professionals and how use cases relate to customer-centric orientation.

c) You are a software designer and are supposed to review the use case description presented in Figure 1. Your tasks are to

- record what mistakes and omissions there are in the description

Name: Registering the usage of the voting right of the voter
Summary: The election authority registers that the voter has started to vote.
Actors: Voter, Election authority
Precondition: Election authority has checked the identity of the voter.
Basic sequence:
Step 1. The system retrieves the information related to election from the database.
Step 2. The system shows the list of polling stations
Step 3. The user selects polling station where (s)he is making registrations.
Step 4. The system loads the information related to election: information of candidates and voters.
Step 5. The election authority enters the identity number of the voter.
Step 6. The system retrieves the information related to the voter from the database: name, address, and voting status.
Step 7. The voting status of the voter is checked.
Step 8. The system initializes the electronic voting card and the election authority gives the electronic voting card to the voter.
Step 9. The system informs the election authority that the voting status of the voter is now updated to be "started".
Postcondition: The voter has been started to vote, this information has been registered into the system and the voter has the electronic voting card.
Exceptions:
Step 5. The system informs the election authority that there is something wrong with the identity number.
Step 7a. The system informs the election authority that the voter has already casted a vote. It shows the following information: voting status, voting date and polling station.
Step 7b. The system informs the election authority that the voter has already started voting. It shows the following information: voting status, voting date and polling station.

Figure 1: Use case description to be reviewed

2. Testing. Answer a and b

James Whittaker[1] (2000) described core challenges of software testing in his article "What is Software Testing? And Why is it so hard?" He presented the challenges in terms of four areas (or phases) of software testing:

- Modeling the software's environment
- Selecting test scenarios
- Running and evaluating test scenarios
- Measuring testing progress

a) How can Extended use case (test design pattern) technique be utilized for solving or mitigating the challenges of testing?

b) How could modeling quality requirements using QUPER (Quality PERformance model) help solving some of the challenges?

3. Domain modelling. Answer a and b.

The (part of) domain model in Figure 2 is related to the use case description in Figure 1.

a) Propose appropriate corrections to the domain model in Figure 2 and justify them briefly.

- Correct errors and improve the consistency with the use case description.

b) Modify (or add to) the domain model to express

- A candidate is a member of a political party and is a candidate in one electoral precinct.
- Each voter belongs to one polling station, where the voter can vote.
- There are multiple polling stations in each electoral precinct.

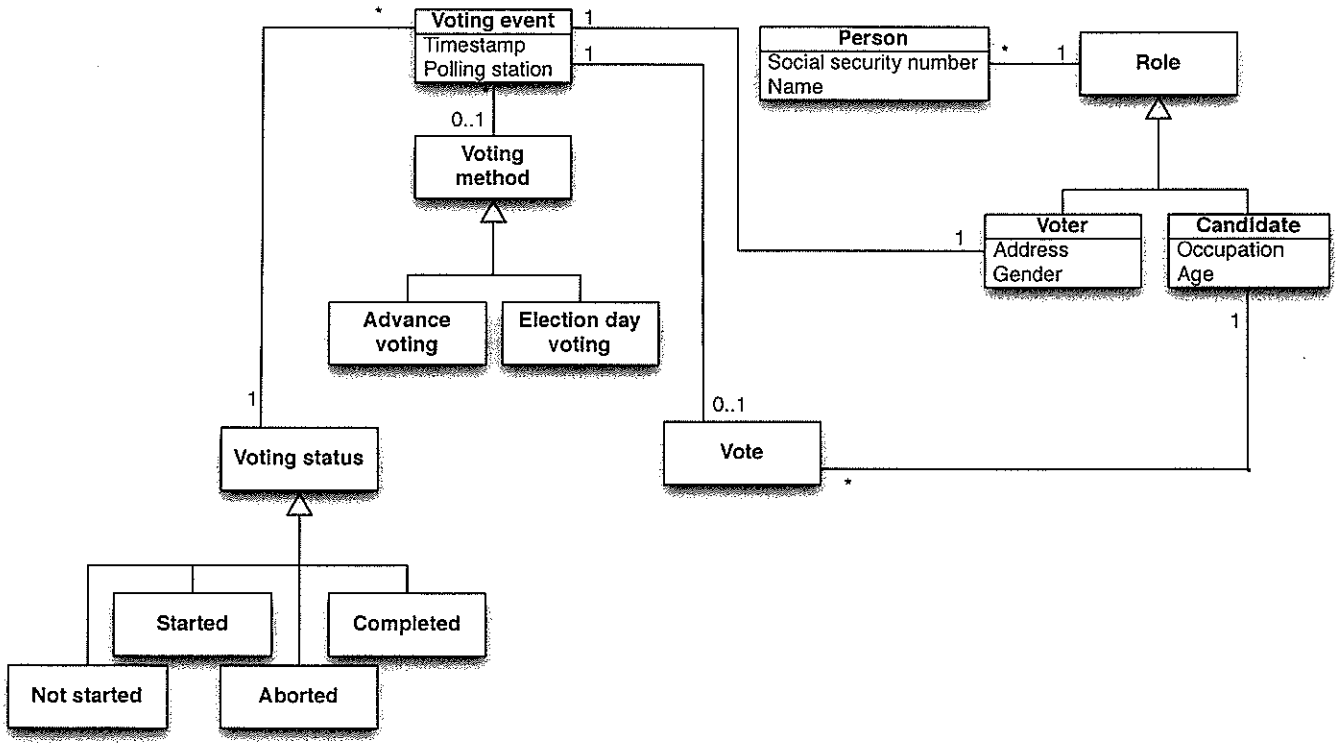

Figure 2: Domain model to be reviewed

[1] Whittaker J, (2000) "What is Software Testing? And Why is it so hard?" IEEE Software, Vol. 17, pp. 70-79.

4. Software architecture.

Use the following problem description to **evaluate all four potential architectures** (Figure 4 a,b,c,d). **Identify** potential **benefits and problems** for each of the proposed architectures. If some of the same observations apply, please remember to repeat these in all cases. **Explain all assumptions** that you use in your analysis.

Problem description: You are working in a small company that is developing a fire detection system. The system has two types of sensors supported. One or more temperature sensors are always installed in the system. Based on the trend in the temperature values the central unit should determine whether a fire alarm should be given. Optionally, a complex and expensive air analysis sensor may be installed. Air analysis sensor produces raw data at the rate of 13Mbit / s.

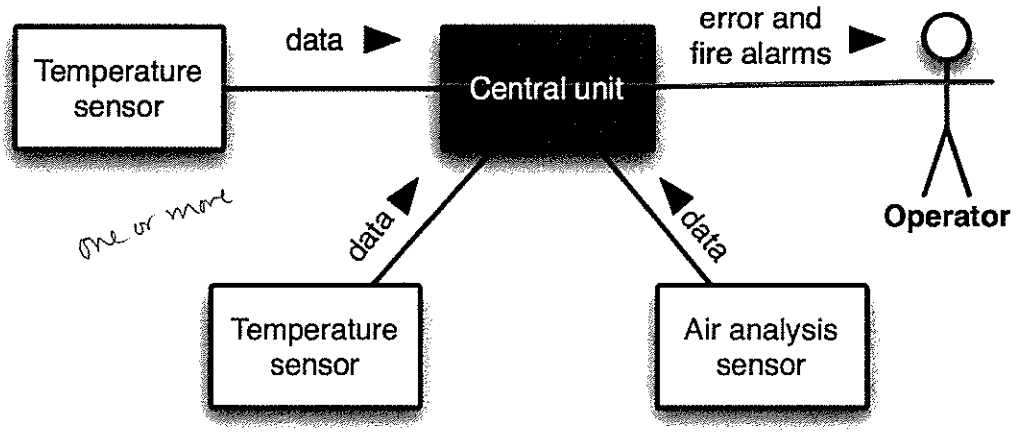

Figure 3: System context

The possible architectures of the central unit to be evaluated: All architecture figures (4 a,b,c,d) represent the central unit. You can assume that if no data store is shown (Figure 4 a,b), then each component will store all the needed data for that component in memory. All dedicated data stores (Figure 4 c,d) contain persistent data. Arrows that do not connect to any component transfer information outside the system. The sensor interface represents a software component in the central unit that reads and gathers sensor values and gives sensor reading onwards to the rest of the system. Data from all the sensors arrive (this is not shown in figures 4 a,b,c,d) to the sensor interface component. For each component, there will be one operating system process. If you find it helpful, you can consider different scenarios: normal operation, system behaviour during fire, and system behaviour when a sensor fails and produces erroneous sensor data.

Figure 4 a)

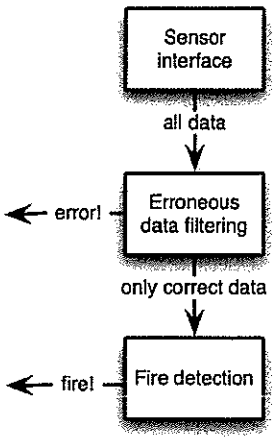

Figure 4 b)

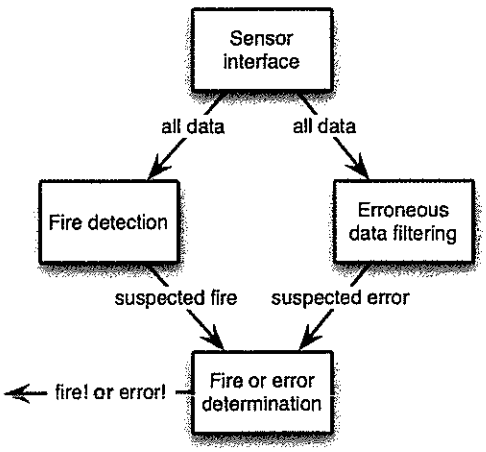

Figure 4 c)

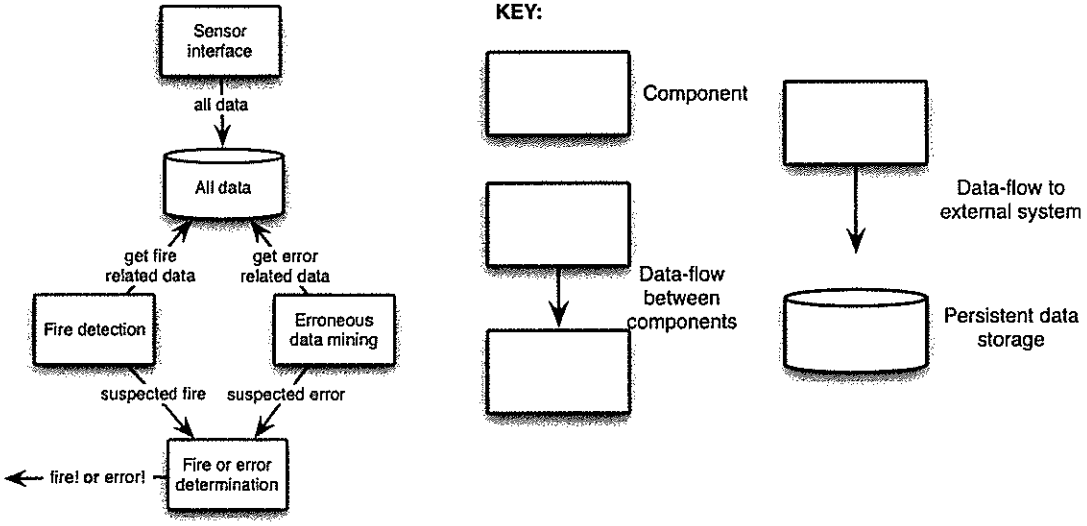

Figure 4 d)

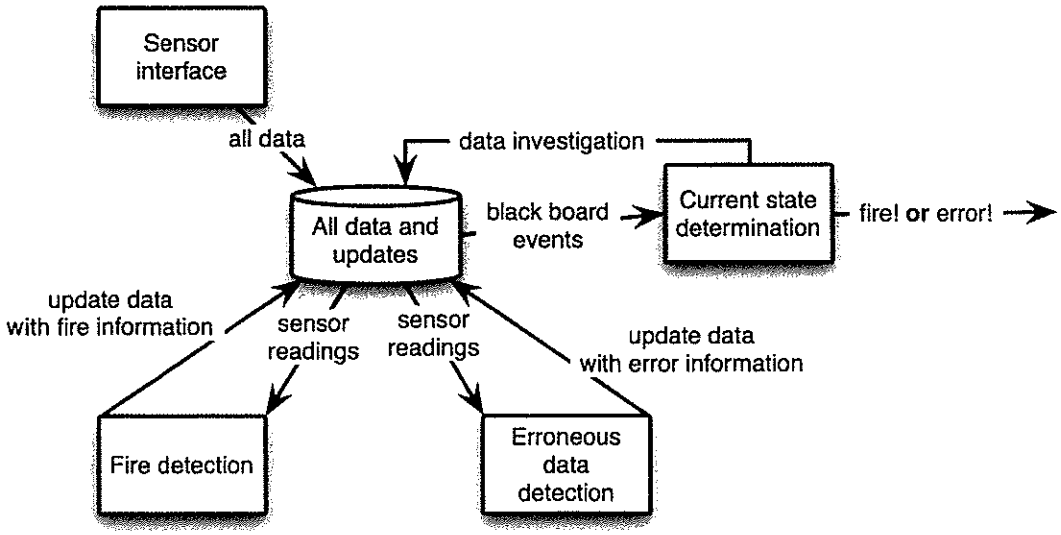

Figure 4: Software architectures to be reviewed

Part B of the exam (to be returned *later*)

This part of the exam you should take with you, complete on your own and return **by the 26th of January to dovehouse compartment 7.**

We expect your answer to be an essay with a good structure, reflecting your own experiences from the exercise with the ideas presented in the articles, and justifying and explaining your answer.

The expected length for the essay is 2–3 pages. The maximum points from the essay are 6 to be added to the exam points.

Do either a) **or** b):

a) Tasks for students who did the exercise of the course **as team work.** Note that these tasks are to be done on your own.

Eisenhardt et al.[2] (1997) have discovered six tactics that successful management teams apply for managing interpersonal conflicts.

- Analyze your team's activities during the exercise from the perspective of these six tactics
- Analyze how the value skills (Denning and Dunham[3] 2001) and the six tactics (Eisenhardt et al.[2] 1997) are related
- Give practical guidelines on how to implement and support the six tactics and value skills in practice in a software development team

b) Tasks for students who did the exercise of the course **as individual work**

Eisenhardt et al.[2] (1997) have discovered six tactics that successful management teams apply for managing interpersonal conflicts.

- Analyze how the results of the exercise would have changed if you had done the exercise as team work and applied the six tactics in your work
- Analyze how the value skills (Denning and Dunham 2001) and the six tactics (Eisenhardt et al.[2] 1997) are related
- Give practical guidelines on how to implement and support the six tactics and value skills in practice in a software development team

---

[2] Eisenhardt Kahwajy and Bourgeois (1997) "How Management Teams Can Have a Good Fight", Harvard Business Review, Vol 4, pp. 77-85.
NOTE: This article is now available in Noppa.

[3] Denning P J and Dunham R, (2001) "The Core of the Third-Wave Professional", Communications of the ACM, Vol. 44, No. 11, pp. 21-25.
To find this article, use for example Google Scholar.