

# Assignment 1: Week 35

HJSK and MGAE

September 2017

## Exercise 1

In the following description, explain when the term account is used as an application domain concept and when as a solution domain concept:

"Assume you are developing an online system for managing bank accounts for mobile customers. A major design issue is how to provide access to the accounts when the customer cannot establish an online connection. One proposal is that accounts are made available on the mobile computer, even if the server is not up. In this case, the accounts show the amounts from the last connected session."

text

## Exercise 2

A passenger aircraft is composed of several millions of individual parts and requires thousands of persons to assemble. A four-lane highway bridge is another example of complexity. The first version of Word for Windows, a word processor released by Microsoft in November 1989, required 55 person-years, resulted into 249,000 lines of source code, and was delivered 4 years late. Aircraft and highway bridges are usually delivered on time and below budget, whereas software is often not. Discuss what are, in your opinion, the differences between developing an aircraft, a bridge, and a word processor, which would cause this situation.

text

## Exercise 3

### 3.1

Explain the differences between software development and software engineering.

text

### 3.2

Describe the history behind the term 'software engineering'.

text

### 3.3

Explain the expression 'there is no silver bullet' in relation to software systems.

The phrase is an analogy to werewolves and how a silver bullet can magically eliminate this monster. Brooks argues that the existence of a silver bullet in software development is unlikely to exist due to several complexities in this field. Firstly, the rate of which technology is advancing far outpaces the advancement of tools and practices in the hands of developers. The use of both high-level and OOP languages may have significantly increased software productivity, but fundamental elements of development continue to cause problems due to their nature.

A software entity is inherently complex because of the many states it can take, which makes it hard to understand and test in depth. The discipline is also considerably more abstract compared to other scientific fields due to the rules being made by people and not nature. Lastly, it has no physical structure which restricts the number of ways our mind can understand it. [Frederick P. Brooks, 1987]

Because of these reasons, there might not be a key or silver bullet to the complications of software development.

referencetest [Bruegge and Dutoit, 2010]

referencetest [Nagel, 2016]

## References

Bernd Bruegge and Allen H. Dutoit. *Object-Oriented Software Engineering*. 3rd edition, 2010.

Jr. Frederick P. Brooks. *No Silver Bullet: Essence and Accidents of Software Engineering*, April 1987.

Christian Nagel. *Professional C 6 and .NET Core 1.0*. 1st edition, 2016.