Assignment 01

Anton, Selma, Bertram

September 2022

Generics

In the first version of the 'GreaterCount' method the only type constraint applied to T is that it must extend the interface *IComparable*. In the second version there is an additional constraint which says that T must be an instance of the class U and that U must extend the interface *IComparable*.

Exercise 1

I want a version control system that records changes to a file or set of files over time so that I can recall specific versions later. This system should work on any kind of files may they contain source code, configuration data, diagrams, binaries, etc.

I want to use such a system to be able to revert selected files back to a previous state, revert the entire project back to a previous state, to compare changes over time, to see who last modified something that might be causing a problem, who introduced an issue and when, etc.

Figure 1: Noun/Verb technique - Nouns are marked in red and verbs in blue

1. All of the nouns and verbs describe a desired solution and are therefore part of the solution domain.
2. In the implementation libgit2sharp there is no such classes as *File* or *State* even though these nouns were found during the Noun/Verb method. The classes are possibly omitted since there is no need for creating instances of these classes.

Exercise 2

The Corona Pas App is an *Interactive transaction-based application* because this application execute on multiple remote servers and computers, and are accessed

by users on their own mobile devices.

Git is a *Stand alone application* because it runs all the necessary functionalities on the user's personal computers.

Exercise 3

The corona pas app is customized software since it was commissioned by the Danish Government and developed for this purpose in particular. Git is a generic product as it is not created for a specific customer but as a platform for whatever a user can think of using it for.

Exercise 4

Acceptability is important for the corona pas app and for the insulin pump as both applications have users with varying technical skill level and it can have big consequences if the users can not figure out how to use it. Especially concerning the insulin pump system.
Git does not have the same requirements for acceptability as it is not an app that needs to be used by everyone.
For all three applications there is a certain need for both dependability and security. The system that relies most on dependability is the insulin pump as it could be a life-or-death situation if the pump failed. Security is most important for the corona pas app as it handles personal information.
The corona pas app is dependent on efficiency for ease of use. There can not be any delays in administering insulin to a user of the insulin pump so efficiency is very important. For Git efficiency improves the user experience but is not crucial for usage.
The needs of the users of both the insulin pump and the corona pas app does not change over time so the need for constant maintenance is not necessary. On the other hand Git is ever changing and the use of it evolves constantly.

Exercise 5

1. Gitlet consists of only one file and therefore only one component. An architecture describes the relations and structure between multiple components and therefore there is limited use of architecture for Gitlet.

2. To infer the architecture of Git without any documentation, an extensive analysis of the source code of git could be done. This way you could map the different data structures and classes together to get an overview of the architecture.

3. Gitlet has the same functions as Git i.e. push, pull and clone. The difference is that Gitlet implements all these functions in one single file whereas Git has multiple files.

4. As Gitlet was written for the purpose of making Git more easily understandable, acceptability has been a priority. As there was a need for Gitlet it speak to the fact that the acceptability of Git has been more limited and targeted towards a smaller demographic.
   For Git dependability and security is key. It is used as a workspace in every software and IT company around the world. Downtime, security breaches and loss of data can cause enormous economic damage to a company.

Exercise 6

1. The issue with the program implemented at the veteran hospital was that the users of the platform was not probably informed about how the system worked. The program did not display any error message if the form was not filled out properly and the form then ended in a queue no one checked. The issue with Sundhedsplatformen was that after a change in the code the platform did not work as it was meant to.

2. If the system at the veterans hospital had warned the doctors that the form was not filled out as needed or the doctors were educated on using the system the problems could have been avoided. If the 'unknown queue' had not been implemented and all requests had entered the same queue, filled out properly or not, all requests would have been processed. To avoid the problems with Sundhedsplatformen there should have been more extensive testing before the new implementation was released.

3. From a resource perspective it would be cheaper to remove the 'unknown queue' all together and display an error message when the form is filled out incorrectly. As a software engineer you would not solve the problem with Sundhedsplatformen by hiring an IT department to report the errors to but test more extensively before release.

4. Ethical dilemmas arising when developing software for the healthcare industry is firstly if any patients would be harmed by flaws in the software, who is to blame for such an incident and is it ethical to profit off of people in need of healthcare services.