



# Final Report

## Team 7A.2

<u>Name</u>	<u>Contributions</u>	<u>Signature</u>
Grzegorz Lindert	The final report and program were written equally.  Program contributions can be seen on GitHub.	
Pawel Szymczyk		

### 1. Introduction

This document is a retrospective summary of a yearlong software project for the program called “ComPChecker”. The document is subjective opinion of the authors.

### 2. Coursework Overview

An aim of the coursework for Software Engineering unit was to develop a range of skills including: level of team – working, planning, negotiation, initiative, leadership, project schedule, and report-writing skills. All students were divided up to 6 people groups. Each group had had to figure out own idea on the software they want to make and realize it using available tools. In addition, to show a progress of the work, the team had had to submit series of the documents: “Project proposal and project plan”, “System Requirements Specification Document”, “Design Documentation”, “Prototype demo”, “Testing Documentation”, and “Final Report”.

### 3. Group Work

At the start, the project had six members. In the first meeting in the classroom, the members were introduced to each other and have started to understand other members’ hobbies and skills that could be used in the project. In our first meeting outside of the class time, we continued to get to know each other and started to think about the project. This required brainstorming about the idea, roles, strengths and weaknesses in order to distribute roles to each member according to their ability. The roles assigned involved both technical and non-technical. At this point, Tomasz was set to be group leader which was meant to keep track of deadlines and ensure the work is up to a good standard. In addition, the group agreed to meet every Wednesday after Web Script lecture.

The next meeting led to the agreement on the application and its purpose. The first idea was to develop a website, however, not many members had experienced in it. Therefore, the group agreed on a desktop application for creating desktop computers. The language of the application is Java, as all members have been familiar with it. SQL was used to create a database for the application and NetBeans was used to program the application. At this stage, there was a good atmosphere in the group and overall agreement on the project.

When the group started to work on SRS, potential users were interviewed to understand their need and requirements. This stage helped to prioritize features helpful to users, in order to complete the project within the timescale. However, the scale of interviews could have been greater as it would give a much better idea of the needs of the user. At this stage, the group crisis has begun. This was because other team members did not seem to listen to the advice or see the point of other members of the group. Only a few members could decide was needed to be prioritized or included in the program even though not everyone agreed on it.

The submission of the Design Documentation which led to complete breakout of the group. Initially it was agreed to work on the application over the Christmas break, however, none of the members did commit any work over that period. Mostly because the deadline was in the middle January and therefore there was still potential time to do the work after the break. Another thing was that members of the group were given task without agreeing on it, which led to complaints about assignation of work.

The next submission of the prototype was in light of the group split. The leader has decided to quit his position. This wasn’t a problem as he never did what he was supposed to do. On the other hand, there wasn’t any sort of communication in the group as three members have decided to start their own application behind the back of other members, still pretending to be part of the initial group. When they were asked to do some work, they never had time. This was unfair as it was about three

or four days before the deadline when they finally decided to split. At this point, the amount of work doubled on the members of the team, especially because the deadline was approaching and the application wasn't to the expected standard in order to produce prototype video demonstration.

In the end, the application was written by two members of the group as the third member did not do any work as he had some personal issues. By all means, this reduced the workforce even more after the split. The group after the split have never stacked to the project schedule due to it being designed for 6 people working rather than two. The only deadlines that were met, were coursework submission deadlines.

### 3. Issues

The main issue was the lack of communication as the group which led to the split. However, the split meant the program could be written as the members wanted it to be and how it was planned at the beginning.

The major issue was GitHub. When two people used to work together at the same file and then committed the changes. One person had to revert commit in order for the program to work again. By all means, the person had to start to work again on that part. This led to losing crucial time for developing the application. We have resolved this issue by specifying who works on what file of the program so that we don't have sync failure.

Another major issue that also led to the split was the fact that half of the group wanted to do code within a number of classes rather than a layered interface. This group has decided to do layered interface which three classes and three forms. This seemed to be the simplest approach to the project without complication it too much and creating unneeded files.

Another issue after the split was with one member not cooperating which led to another internal problem. He promised to do work and then always found an excuse for not doing it, which can be seen in a number of commits on GitHub. Eventually, he agreed not to assign him any work and to submit without him.

### 4. Conclusion

Overall, this project was difficult as it consisted problems both personal and technical. It taught a lot about weaknesses and how to work on them. But the real valuable things what we learn as the team. In fact, that it is communication and trust that are key to the success, and a lack of them, gives low results and breaks the team. As the team people should work together for the same goal rather than working for themselves, which doesn't lead to any outcome in the end.

### 5. Appendices

Source code: <https://github.com/fccc0239b79/PChecker>

Video demo: <https://www.youtube.com/watch?v=nmEJpd8EkEI>