Dominik Schmidtlein 100946295

Reflection 1

The trial project had a lot of positives that would be nice to see in the term project. For starters, the scope was broad enough such that were busy until the end, and that the result was interesting. All the same, the scope was narrow enough that we were able to finish all the goals on time.

Another positive was the team’s ability to use the resources to their full extent. On the piface, the team used both the buttons and the lights. On the gert, board the team used the lights, even turning lights on individually which was a challenge on its own. The team was even able to use sound on the raspberry Pis, in order to add to the ambiance of the game.

The lesson to be learned from the trial project is that we were able to use the technology to its potential and even push the boundaries of what was expected. We would like to see this repeat in the final project.

As far as using the trial project as a learning experience goes, the project has 1 major component and a few simple components which made task division difficult. The Ping Pong game had 1 server which contained all the logic and made all the decision, as well as communicating with all the other devices. Then there were 2 players, whose responsibilities were merely to send a message upon button click. The final device waited for messages at which time it played music. Therefore, there were 3 devices whose configuration was less work than each person was entitle to, and 1 device which was responsible for almost the entire project. The problem was that each person was given the responsibility of programming 1 device. As a result, 1 person ends up doing more work and others, who would like to contribute more, cannot help. This is a situation that we would like to avoid for the term project.

One solution, for better task division, is to have multiple people working on a part at once. By using pair programming, the development of any given part will be faster, and multiple people can contribute. Pair programming also produces better code, which is always useful. Therefore, instead of having 1 person do the server alone, 2 people could have programmed the server, while the other 2 are responsible for the sound and player devices. The result would be better code all around, as well as better task division.

From the trial project, we learned how the individuals in our team work. One part of the teamwork that was smooth was our ability to divide and conquer. Even though our choice on how to divide the tasks was poor, we all got our assigned tasks done, and at the end the project came together smoothly. On the other hand, we did notice some friction when deciding how to program specific functions. The problems became apparent when team members attempted to work on tasks that were not assigned to them. Since everyone can have different opinions on how something should be programmed, it is safest to have trust in the teammates and let each member complete their own part.

The documentation was completed efficiently since the parts were divided and then each member could work simultaneously. Therefore, instead of 1 person having to work on the entire report, each person only had to work on 1 quarter of the report.

In term of coding, the hardest part was always interacting with the hardware. For example, getting the lights on the gert board to work independently was time consuming. The difficult part was finding effective documentation which explains how each part works. When the documentation is available, the implementation is very easy. Therefore, the only thing we need to have more of is access to effective resources that can accelerate the process of learning how to use the hardware.

Although I have already seen our mark, I will approach this question as I would have, prior to seeing the marks. The first evaluation is problem identification. I feel we clearly identified that we were making a ping pong game for 2 players, which plays music when someone loses. Therefore, the scope was clear and our objective was focused.

The proposed solution evaluates the balance between pushing the limits and being realistic. The solution was broad as we attempted to use the gert board lights as well as the sound module. On the other hand, we were able to finish all of ours goals without cutting any corners. Thus, the solution is well though through and of the right complexity for the trial project.

The implementation evaluates how well the solution works. Since our solution worked without crashing, it met the expectation for implementation. However, implementation also evaluates the quality of the solution. There were certain parts of the solution that did not produce robust reusable code therefore, the quality was not perfect.

The most testing that was used in the trial project was experimental testing. Therefore, not all edge cases were considered, thus, there could be circumstances under which the solution would crash. Once again the testing was good but not excellent.

The professionalism was not evaluated for the trial project. The individual and team work was also not evaluated for the trial project.

In summary, 2 of the sections, problem identification and proposed solution were excellent. The other 2 sections, implementation and testing were graded in between good and excellent. Therefore, for my personal evaluation, the grade is an excellent-.