**Self-evaluation of my performance during the project**

I would evaluate my performance in this project to be very good. I have done everything according to the *Collaboration Agreement* and tried my very best at all times. Asked for help when I needed it and help when asked to. If I had more hours to work a given week or day, I would find something that needed to be done and do it. And more often than not I would end up working more then I needed, this contributed to me closing in on 165 hours of work earlier than the other team-members. This happened in a greater degree when the end of an iteration was coming up. I tried to distribute my workhours evenly, across work areas (Programming, Documentation, etc.) and also in hours per week/iteration, but this was hard because it was more to do in the second and third iterations. In the end I had a little over 40% on programming and a little under 25% on documentation. These two were the main categories were everyone worked. The reason as to why I have more programming is simply that during my last project I did a lot of documentation and felt like I would get the most out of this project by for the most part do programming. I feel like I have got a lot out of this coding experience. Also, there is the fact that most of the documentation was at the end of the project and when the team started on this, I was already low on hours left to work. And the documentation for this project did also take up a smaller portion of workhours spent on this project in total.

According to the teams *Risk Analysis* if someone became sick, absent, etc. the rest of the team would compensate. I tried my best to compensate for other team-members who were absent or sick.

Unfortunately, there were much more sickness within the team during the project than anticipated, this also contributed a great amount to me closing in on the 165 hours limit faster.

In the end I did end up with 165 workhours. There are many factors which contributed to this, I have mentioned some of them earlier, and will elaborate a bit more now. One of the bigger reasons that I worked 165 hours is because of poor time management and communication within the team. During the first and most of the second iteration I was under the impression that everyone else had worked equal amounts in hours (taking in to consideration that two of the team members were sick for about a week during the first iteration, and therefore a bit behind, but catching up), but I did not check their timesheets. However, there was one team-member that was behind with no good reason. This became quite clear when we approached the end of the second iteration, when another team-member and I “suddenly” was far ahead of the other team-members in hours. Note that this is also due to the fact that I was working on the things that needed the most work before the MVP, hens this became partially my responsibility since it was only the other team-member I worked with and I who had the knowledge, at that time, to finish the code (javaFX, frontend) needed for the MVP.

In the aftermath of the second iteration I informed the rest of the team about my hours, and we decided that I would take a “break”, or at least work less for a few weeks, so that I could work the needed amount on things I had done before when this became more necessary later in the third iteration. Due to this I worked less in week 13 and 14. This worked for a while, but I ran into the same problem later because the team had their own deadline a week before the real one. I then ended up working a lot again to try to reach this deadline, but some of the other team-members did not share my ambitions for being done by this deadline. After week 15 I was already quite close to the 165 hours limit. So yet again I worked less, this so I could help with things I was more familiar with, so the team was the most efficient, because at this point the team was, in my eyes, far behind where they should have been. The end result of this became that I on purpose enter week 17 with 5 hours left of work. These hours were supposed to support the others and do small and quick things that would slow the others down. I ended up with using up my last hours the days before the actual deadline.

I shortly want to emphasise that the team manged to have a smoother and more efficient workflow during the third iteration even when the areas of work was much more entangled and it in general became more demanding to organize what to do, in what order, how many should work on a task, and especially how long it would take to do (this since many of the tasks we had were new to us, and we did not have a frame of reference).

At the start of the third iteration we began with daily status report meetings, as a counter measure against the poor communication due to the COVID-19 lockdown. This helped a lot with the communication within the team, and it was also easier to keep track of what other people were doing. The team coordination got better as the project went on, and by the end I would say that we had good communication within the team.

All the team members were friends prior to this project, this came with some benefits, but also some difficulties. The only real benefit was in the start, were we had a “jump start” compared to other teams that had to go through the “get to know each other” part of the team process, and the fact that you know you can tolerate the people on your team. However, from my experiences with this there are more downsides. It is harder for most people to criticise the other team-members in this situation, this became a small problem for us, but nothing major. I did push for some quite harsh and strict roles in the *Collaboration Agreement*. To give team-members something specific to point at if they needed to remind someone else about the terms of being part of our team. Nonetheless, we had some difficulties following this agreement. If a team-member did something the rest did not like, but it felt petty or passive aggressive to say something, so they rather turn a blind eye. (e.g. not coming to daily meetings at a reasonable time.) At one point the team considered to kick out one member, but the member was given more chances than if we had not been a group of friends. The “kicking out” part was then procrastinated so long that it was even more cruel to kick the member out. So, it became even harder, and the team ended up taking a decision for the best of the project. We decided that the team-member in question would be more of a resource than a burden for the remainder of the project, and after the team-member became better at the things we talk to them about, it was left at that. I do however think that the team-member caused a lot of delay in the progression the team had, but the team again compensated by cutting functionality to get a decent product in the end.

My role in the team was team leader, I would evaluate my performance in this role as quite good. I have fulfilled all the specific tasks I was supposed to, such as writing and sending out meeting summons, leading the meetings. I also feel like I have had good control over all deadlines, informed the team about new information, so that everyone is up to date on everything. The only thing I would say I have failed at as a team leader is during the first iteration and the first half of the second iteration when I did not have enough insight into what all the team-members were doing and that this hurt the overall product in the end. However, I feel that once I noticed this problem, I became well informed about what each team-member was doing, this became much easier when we started with our daily meetings. I have been an objective and fair leader, we have had a democratic decision system, so other than my additional tasks I have functioned as a normal team-member.

I have learned many new things and delved even deeper into knowledge I had prior to the project. I have become quite familiar with JavaFX and what is possible to accomplish with this, I also appreciate tools like SceneBuilder and have learned when it is best to use them, and when I will benefit from doing it by myself to make things easier for me later. GitLab is an awesome tool for keeping track of what others are doing or planning to do, and also to efficiently share code. I was already quite familiar with what was expected of the team when it comes to documentation, but I feel that I got an even better grasp of the importance of it, mostly because of the cheer size of this project, this also made the documentation in general seem more needed then on earlier smaller projects I have been involved with, were it the documentation just seem like a waste of time. However, I think I have learned the most about the importance of teamwork and collaboration, this was especially made clear to me when it was impossible for me to fix things on my own and had to rely on others contribution to get things done in time. To have good communication within the team is also extremely important, otherwise the product will suffer from this. I personally feel like it was more problems than benefits from being a group of friends that became a team. Nonetheless, we did lower the risk of team-members getting hostile against each other, by being friends.

I want to point out the importance of dealing with problems when they occur and not procrastinate or wait and hope they disappear on their own, it is in the team’s best interest to always discuss any displeasure to keep a healthy work environment, we were a bit procrastinating because we wanted to avoid heated discussions with our friends. Try to be friends during free time and colleagues when working. This is very hard, but everyone needs to know that business is business. I have learned that I work much more efficiently when working with a partner rather than alone or in bigger groups. I have gained a lot of experience about how team projects and this type of agile project structure works and have definitely noted things that I will take into consideration or focus more on in projects to come.

To conclude I felt that I contributed a lot to this project. I did not waste my hours; they were spent efficiently. I think I was a successful team leader, and I felt that the rest of the team respected my role. I feel like I have learned a lot during this project, and that most of it will become highly relevant later in my studies and in my working carrier. Overall, I would say my performance on this project has been very good. The teamwork was not that good during the first half of the project, but the improvements that came with daily update meetings and such made it much better, and by the end of the project I would evaluate it to be quite decent.