Behind the Scenes of the Project

# Introduction

As our mini-capstone project comes to an end, it is essential to look back and analyse the process in this postmortem analysis. The objective of this project is to develop a condo management website and its companion mobile app for potential buyers, condo owners and condo management companies. The core structure of this application is an Enterprise Resource Planning (ERP) system, which encompasses various functionalities such as user profile creation, a reservation system,a management system, and a financial system.

Our main goal was to offer the client and end users a web and mobile platform that will allow condo management companies to use our software to manage their condo buildings, for condo owners to manage their respective condo building and for public users to be able to access information on their condo units through this app.

When reflecting on this project’s outcome, we were proud of our accomplishments but still believe that there is room for improvement. Among various issues encountered, were our bad time management. Throughout the semester, team members were busy with their other classes and projects, which made it difficult to coordinate when to finish our tasks. We also had issues involving understanding the requirements. However, we were able to tie up any loose ends before deployment and prepared the product presentation as hoped in the sprint 4 retrospective.

After filling out a retrospective board anonymously, we saw that there was a mutual agreement concerning our organisation. Although it has significantly improved since the first sprint, there is still some confusion as to what needs to be refined and reworked on. We also encountered a few merge issues on our GitHub as well.

# What went wrong

## 1 - Lack of communication

Throughout the project, team members were often busy with assignments, other projects, and exams. Once again, everyone had very different schedules which made it challenging to assign tasks as we were often unavailable to hold a meeting altogether with all the members. The impact of this was that we encountered unnecessary merging issues since team members would start working on tasks without notifying others which caused a lot of time to be wasted trying to fix resulting merge and build conflicts.

We addressed it by holding small meetings with whichever members were available at the chosen time and so we were able to assign most of the tasks. However, it would have been better to hold more frequent meetings to update each other on what is left to be done.

## 2 - Time management

Throughout the project, the team struggled with time management, because, as mentioned in the last point, every team member has very different schedules and responsibilities. We all had assignments and group projects in other classes as well. This made it so that we often left our task to be done at the last minute. As we scramble to get our work done in the last few days of each sprint, we fail to pay attention to small details that end up costing us points that could have easily been avoided. We also ended up having to stay up late trying to fix our errors and make as much significant progress as possible. This could have been avoided had we given ourselves more strict deadlines and held each other accountable.

We addressed this issue by communicating to each other in the last week each sprint what is left to be done. Notably, we make sure to communicate to each team member the tasks they need to accomplish for the sprint. However, it would have been better if we had communicated that from the very beginning and checked up on each other to ensure that progress is made in an appropriate time frame instead of all near the end of the sprint.

## 3 - Better understanding of the requirements from the get-go

Throughout the project, the team struggled with understanding the requirements. As we had bad time management, we often overlooked small details that mattered. As a result, we lost points due to our lack of clear understanding of the deliverable requirements. For example, there were mistakes or missing information in documents in our Sprint 2 deliverables that were not fixed until Sprint 4. We were not careful enough in trying to understand our errors which led to us losing points that we could have easily avoided losing. This could have been avoided had we taken better care and more time to fully understand the requirements as well as asked questions to the client (the TA) earlier on to take those new information into account for our sprint.

We addressed the issue by making a list of questions to be asked to the client (the TA). We also made use of our demo time with the TA to verbally ask questions and clarifications on the expectations regarding our software and our documentations. However, it would have been better to have had a clearer idea of what was needed from the beginning or early on throughout the sprints so that we don’t waste time fixing errors caused by our misunderstandings.

# What went right

## 1 - Teamworking skills were improved

Throughout the sprints, we were able to collectively work together to resolve issues caused by lack of communication. Although we had a rocky start, we were able to more effectively divide our tasks through our newfound understanding of each other’s strengths and weaknesses. Through the project, our teamwork has gotten better and, even though we lacked communication, redirecting our focus on tasks that required immediate attention enabled us to successfully deploy our Web App.

However, it was tricky to figure out at first how to approach the existing issues in our codebase, we were nonetheless able to communicate and resolve the various inconsistencies.

## 2 - Flexibility and adaptability of the team members

Throughout the project, team members were willing to challenge themselves and take on tasks even though they had little knowledge of the technology required for that part. Moreover, team members were willing to get together to put collective effort to help each other out with the remaining tasks. Other members were also willing to give tutorials to teach other members how to do their work.

The team’s flexibility and adaptability made it so that the work that needed to be done was ultimately finished in the end despite our lack of communication and organisation at the beginning. However, had we been more flexible at the beginning, we would have encountered less difficulties during this sprint.

# Lessons Learnt

## Positive

Through this mini-capstone, we learned the importance of teamwork in a group setting project. We started off working on our tasks individually with little communication and teamwork. This resulted in a lot of conflicts when it came to merging our code branches. There were conflicts in our frontend designs as well for which we had to fix later on in the project. We were able to recuperate through better teamwork as we learned to work together more and share responsibilities instead of individually working on our tasks.

Additionally, we were able to learn new technologies for the sake of this project. For example, we had to look into cloud deployment methods and services. Some of us also had to learn Next.js for this project. Therefore, there were definitely some useful takeaways that came as an outcome of this mini-capstone.

## Negative

On the other hand, some negative learning outcomes would be, as mentioned before, poor and unclear understanding of the requirements from the beginning, time management and lack of communication. These were all things that we needed to work on and that we still need to improve on.

Our misunderstanding or lack of in-depth understanding of the requirements led to wasted time fixing our errors, had we just been more careful and paid more attention from the start. It also led to our grades suffering. Our bad time management caused us to always rush at the end of the sprint to get things done which then made the quality of our work suffer. Finally, our lack of communication caused frustration within the team. It was also a factor that influenced our bad time management.

All this to say that, through this project, we were able to learn the importance of these factors in a group project as they can lead to poor productivity and poor quality of the product.

# Conclusion

While reflecting on the project, we realised that we encountered multiple challenges that originated from our lack of communication, time management and poor understanding of the requirements. However, we tackled these problems by assigning subgroups of members to certain tasks to ensure successful deployment. Although our teamwork is not perfect, it has significantly improved by trying to communicate more often to the best of our abilities. Things worked out better in the end compared to previous sprints, since we established earlier on what our vision was for what was to be completed for the end of the project. We got to refine the rough edges and we were able to bond a bit more which improved the team dynamic.

Looking back at the various sprints and resulting product, we have learned many lessons. Notably, the importance of communication, organisation, and teamwork are crucial when it comes to software development, in a team-based environment. In regards to the organisation of the project, the importance stems from the need to prioritise tasks and be realistic about what can be accomplished by implementing effective communication to ensure smooth delivery and deployment. By applying the lessons we learned, we will tackle future challenges by being more informed and committed to developing the best product possible. We will also be able to carry on these newfound skills in future projects.