With the unfolding of the Integrated Masters in Informatic Engineering, I had the chance of experiencing the first iteration of the curricular unit “Large Scale Software Development”, where, together with the rest of my class, we developed a product for our client, Filipa Barros, with the theme “Warehouse of the Future”. In this project, we were tasked with first figuring out what exactly was our role in the project, by listening to the client and deciphering what were the components of the product and how we would start developing it. After having a very rough overview of the problem at hands, we devised four problems to be tackled, our three “verticals” and one “diagonal”, since one of the problems affected all the others, it couldn’t be developed as completely independent from the other three. My team decided to work on the “diagonal”, the creation of a Data Lake for the warehouse, where all the data would be stored and would receive real-time updates, as a way to mimic the warehouse through its data.

This was only the second time I had to deal with a costumer on a project of a certain scale, so I started by trying to specify every detail and obtain the most amount of information possible from our client. By the end of the first reunion, we grasped the overview of the project and were motivated to work on it. Due to being the person that asked the most questions, my team decided to designate me as our team’s Product Owner. One thing that we started thinking right away was the architecture of our part of the project. We knew that our part would focus mostly on a database and a simple frontend, so we had a choice to make regarding which DBMS to use, since we mostly worked with relational databases throughout the course, we decided to try MongoDB for a change. The advantage of this is that, now that the project is finished, we had the opportunity to try a new technology, but the downside is that the tool we choose wasn’t the best for the job, which harmed the quality of the final product. So now we know that, when considering a project, we can first check if its going to be worked on as a learning method or as a product, considering the available time, we can make a more thoughtful choice.

The rest of the project went smoothly, with each sprint we had the necessary meetings and planning, using poker cards we decided on each issue weight, something new for me, and the work was correctly distributed. Another first for me were the periodical client meetings. Since our client worked at FEUP, me and the rest of the PO’s didn’t provide enough attention to our client. Our meetings were mostly marked on the same day and since they were presential, we needed to find a place to have the meeting. With that minuteness aside, the presential meetings were a breath of fresh air regarding the current pandemic affairs and all of them were very informational, on one hand because we had always new questions to ask Filipa, as well as the need to update her on the current state of the project at the time, and on the other hand, we had a new perspective regarding clients and how to interact with them, something that is of great value regarding the course we are attending.

For the final stretch of the project that was made during the last two weeks, we needed to merge all of the group’s partial projects into one cohesive product to be presented to the rest of the course year and to our client. Since we worked on the “diagonal”, we had a big part on the final release. We were confident we had made a good enough job in our part, and we hoped we would be working on helping the other groups integrating each other’s parts and not working on any new features. And it went mostly that way fortunately. The only thing that escaped all of the members of our group was a slight detail regarding how many requests were needed to make to our API to be able to get the information needed. After changing this, we proceeded with the rest of the project, creating our classes’ product and presented it successfully.

Finally, I think another strong point of the curricular unit was that we worked with a monorepo architecture, something new to me and to the course, which surprisingly worked well seeing as more than 25 people had access to this repository and there were only a handful of necessary commits to the main branch. All the other additions were correctly integrated and each team had their own issue board and tags.