**Lesson Learned**

During the project we learned the following:

* SpringBoot Framework is a very efficient tool to develop java web applications
* Using Javascript in frontend is easier to handle RESTful server/client APIs.
* Bootstrap is a great tool to design a comfortable user interface where he can interact with the application.
* Application restrictions are better to be handled in DB. It makes the application faster, secure and efficient.
* Having a pre-installed Github desktop works better than using the Github browser because using the Github browser has one limitation: you can work with your remote Git repositories - not with your local ones.
* Be aware to push changes to the branches first and then merge into master to avoid conflicts.
* Azure data studio is a good substitute for MS Management Studio since working with MS Management Studio on Mac Systems requires overhead to have a docker installed.
* Websockets are a better way to make communication between frontend and database work, but it can be difficult to implement and requires some time.
* Working at a common work location is better to improve communication and face technical difficulties.
* It's important to have a direct communication channel between different working groups, like frontend/backend.
* Ajax library has many fetching methods that make the interaction between the client and the server easier and faster to program.
* IntelliJ is the most efficient IDE for java programming as it has many libraries, features, extensions, packages... etc.
* SQL Server is a very competent implementation of SQL. SQL Server has a lot of functionality that other versions of SQL do not have. Furthermore, especially with stored procedures, SQL Server has more competent features in my opinion allowing for a faster and more secure implementation of our database than what would be possible in something like MySQL.
* It’s better to agree on the format of and standardize the responses between the front end and backend.
* Lesson in the sunk cost fallacy: we should not have waited to the last minute to make changes that we should have made long ago.
* It is more efficient to include the codes libraries locally in the project directory rather than link them online.