

Sprint 2 retrospective

During this sprint we divided the work among the team using the planning that we developed on the *Requirements and System Analysis*. The pairs created during this sprint were the following: Pablo Nuñez and James Stockell, Andrés Durán and Alejandro Gestoso and finally Pablo Robledo and Pablo Sanchez. These pairs work in conjunction with each other to deliver the best version possible of the deliverable, but for now let's see in which areas we worked correctly as a unit, in which we didn't and how to change that.

The Good

Through this sprint we correctly delivered concrete and solid tests, we used time tracking tools, we created a useful readme document, we managed to make a good division of the files, we had appropriate communication between the members of the team and successfully delivered all the necessary work for the sprint. Additionally we also divide the project into front end and back end to improve the project management and development.

The Bad

The areas in which we have to put some work in order to improve in the next sprint are the following:

- **Working Schedule:** We must start working on a daily basis in order to stop working overtime to get the project to the deadline.
- **Planning:** We must start using the planning tools at our disposal in order to better the work distribution among the members of the team, such as the *GitHub Projects* tool
- **Java Conventions:** We must follow the java programming and naming conventions in our project in order to better understand it from an outsiders point of view.
- **Github usage:** We must improve our utilization of GitHub, as we created a lot of not necessary branches, did not use the issues tool correctly and did not follow any kind of standard for our commit messages.
- **Useful commentaries:** We must start writing comments in our methods in order for other people to understand them without losing time, and also annotating them with the name of the author to better identify the person who implemented such functionality.

What we have learned

Looking back on this sprint we learnt a lot of things the hard way. An example of this is the importance of working on a daily basis and planning, which we learnt through the working hours of our last week. We have learned that, in order to make the project cohesive and easier to understand, we would have to follow standards that all of us could understand and use. We also learnt the usefulness of making comments in our code regarding what it does and who did it, due to the fact that it helps us to go faster through the creation of tests and implementation of related methods. Additionally we also learnt that to better work as a team, standardising commit messages and useful branch management were a must for collaborative programming, as otherwise we end up with a lot of commits which we can not differentiate, as well as many stalled branches in the repository.

How to Improve from it and future work

For the next Sprint we would like to implement the following changes to improve the development of the project in the following areas:

- **Git/Github:** Starting to use a standard notation, as well as unifying the language we write the commits in, should be a priority. Moreover, using descriptive names for newly created branches and deleting old ones that are lying around should be done. Opening issues on GitHub would be a way to organise the workflow, and lastly learning to better control git would increase general productivity since we are only using the basic tools that such a complete software offers.
- **Project and Java:** We will start by putting everyone on common ground about React and Spring. Next, we will establish the standards to write Javadocs for some of the methods of the project, and apply the conventions for Java programming (lowercase package names, PascalCase classnames, camelCase variables...). We will of course also take care of other assignments like TODOs in our code and implementing and registering design patterns in our application.
- **Planning and Working Schedule:** We will start working on a daily basis and implement a better development plan. For that we will use in a better way the issues section of GitHub and try inceptiving the work among different small sessions during various weeks against big ones near the deadline of the project.

Difficulties we may find

With these changes that we have decided to undertake in our project, there can be different difficulties we may encounter. Taking into account we are going to change the library for front end from Bootstrap to TailwindCSS, there will be a definite learning curve as we refactor and adjust to the new format. As well, ensuring our git repository and code is as clean and as readable as possible can prove difficult, as we would have to break old habits and maintain a consistent format. Additionally the planning and the working schedule may face some difficulty, as some of the members work on a different schedule and all of us have other projects to take care of. As a final difficulty, we believe that it may be tedious to refactor some code, as we have not been using the full potential of the ReactJS library in some cases, sacrificing performance.

Effort of each member

Name	Hours	Sections members ended up working in
Pablo Aurelio Sánchez	37:04:30	Back end and testing, user stories
Jaime Stockwell	50:36:18	Front end, web design and endpoints
Pablo Núñez	36:59:05	Back end, testing services
Alejandro Gestoso	60:11:58	Back end, testing services, user stories, some SQL
Pablo Jose Robledo	47:54:46	Back end, testing services, reviewing of the documentation and SQL
Andrés Durán	128:22:34	Front end, endpoints and back end, web design, SQL