Sprint Retrospective 1

Project: Home Owners Association
Group: 22b

Issue	Assigned To	Estimated Effort Task	Actual Effort	Done (Yes/No)	Extra Notes (if any)
UML Component Diagram	Everyone	3	4	Yes	We managed to have multiple meetings to finalise on the UML component diagram.
Requirement Analysis	Everyone	1	1	Yes	We decided that every single one of us would go through the scenario and then make their own list of requirements and then at the end we union all of them. This way we don't miss out on some important requirements.
Assignment 1 Task 1 Write-up	Everyone	2	2	Yes	We managed to finish this in time. We went through the scenario and managed to use the methods for DDD and come up with bounded contexts.
Context Map	Everyone	2	2	Yes	

Takeaways:

- We could have been more efficient with our work but as it was our first sprint, we took our time to get to know each other and their working style.
- A considerable amount of time was spent on getting the project template set up and making it successfully built locally for everyone.

Adjustments for the next Sprint:

- We are planning to start up with the implementation in the next sprint.
- We know that it's the exam week during the next sprint and we will not be able to put in the required effort but we will catch up for that in sprint 3.

Sprint Retrospective 2

Project: Home Owners Association
Group: 22b

Issue	Assigned To	Estimated Effort Task	Actual Effort	Done (Yes/No)	Extra Notes (If any)
Members are able to create Activities	Kaushik	4 hours	5 hours	Yes	I could have tested parallelly to make sure that the features are working as expected. I also could have added extra checks before adding activity. I also took extra time as I wanted to familiarise myself with the template.
Members are able to leave an HOA	Konstantin	2.5 hours	6 hours	Yes	This particular task took me that amount of time because I had some serious trouble with gradle and couldn't build the project. It was also an introduction for me so I had to look up how things work in the example
When a user registers, the system must check if the address corresponds to the HOA's area	Konstantin	30 min	45 minutes	Yes	Nothing to add here

Full implementation of the users microservice. Including authentication and storing credentials.	Máté	5-7 hours	7 hours	yes	
Voting decision (creating a base for the voting microservice, with Strategy design pattern) +Abstained votes should not be counted +Members should be able to change already casted votes +Members must be able to vote for board members in elections	Giacomo	5-7	10	yes	The "base" for the microservice became most of it
Implementation for Membership Management in the HOA Microservice and test related endpoints + Find user's role + Find whether a user is a board member	Minh	6	8	yes	

Takeaways:

- We could not put in the expected effort this week because of the midterm
- One of the challenges we faced during this sprint was the lack of tests. We did not write any tests in the first sprint, and this will be an area that we need to focus on in the coming sprints. Writing tests is critical for ensuring the reliability of our code, and we will make sure to prioritise this in the future.

Adjustments for the next Sprint:

- We added some tests but we are planning to add more unit, integration and boundary tests.
- As we were not able to put in our 100% in this project during the exam week, we are planning to work extra hours and catch up for the lost time.

Sprint Retrospective 3

Project: Home Owners Association
Group: 22b

Issue	Assigned To	Estimated Effort Task	Actual Effort	Done (Yes/No)	Extra Notes
User is able to create a HOA	Atanas	2h	3h	Yes	Took me a while to be able to understand how to send postman requests with authentication and all
User is able to join an HOA	Atanas	2h	2h	Yes	A little mishap with old code but in the end i got it running
Testing for HOA controller	Atanas	2h	4h	Yes	
Testing for HoaService	Atanas	2h	2h	Yes	
Members are able to participate in already existing Activities	Kaushik	4 hours	4 hours	Yes	I tested them parallely this sprint. I also used Postman to make sure that all my endpoints were working as intended
Testing Activities related implementation	Kaushik	5 hrs	5 hrs	Yes	Testing took some time but the template code helped.

Members are able to add, remove, edit and display the rules of an HOA	Konstantin	5 hours	6 hours	Yes	The models choice was a little bit confusing for me, as well as the way we decided to store the rules
Test the endpoints for the rules	Konstantin	4 hours	6 hours	Yes	I had a huge problem with versions and merge conflicts
Implement checks for votes and options using Chain of Responsibility +Only the board members can propose and vote changes to the rules +An applicant should not be allowed to be in 2 boards at once +An applicant should have been part of the HOA for at least 3 years to be eligible +Members should not be able to vote for themselves +Board members should only be able to keep the role for a maximum of 10 years +Members must be able to apply for board	Minh & Giacomo	8h	10h	Yes	These were all part of the same activity of implementing the Chain of Responsibility Design Pattern
Blocking new rule proposals during a board election	Minh & Giacomo	1h	1h	Yes	

Users should be able to change their username, password or full name.	Máté	4h	4h	yes	I opted to not make the usernames changeable because that would conflict with the databases in the other microservices.
An endpoint to check if a username exists in the User repository	Máté	1h	1h	yes	
All members of the HOA should be able to receive a notifications	Máté	2h	3h	yes	The functionality for notifications is implemented, but their creation isn't automated.
Integration, boundary and unit tests for the users microservice and notifications	Máté	7h	10h	yes	Boundary testing is largely done in the users microservice, checking for the length of user credentials.
Minimum and maximum length for user credentials when registering or changing credentials.	Máté	30m	30m	yes	

Takeaways:

- Testing with postman
- Spring testing is a little confusing
- Should avoid huge MRs as it is tough to review. MRs should be per issue.

- If anyone wants to make major changes, such as changing the name of a class, they should discuss it with the other members of the team