SPRINT 1 RETROSPECTIVE

PROJECT: “Rowing”

GROUP NR: 29A

Student	Task (Issue nr)	Estimated Effort	Actual Effort	Done	Notes
Alex Preda	Assignment 1 Draft	2h-3h	3h	No	
	Added the project template	1h	3h	Yes	We had some issues with the gitignore and gradle.
	Added issue templates	1/2h	1/2h	Yes	
Anca Badiu	#3 - add CI	2h	2h	No	The project structure was not set up well
Stanisław Howard	#1	1h	5h	Yes	Worked on it with Rafał
	Assignment 1 draft				
Rafal Owczarski	#1	1h	7h	Yes	It was problematic but everything is functioning properly now
Violeta Macsim	#4 - User class	4h	3-4h	Yes	Still has some data to be added
	Assignment 1 draft	2h	2-3h	No	Consult TA’s feedback
Kayleigh Jones	#7 User creation API	4h	4h	Yes	

MAIN PROBLEMS ENCOUNTERED

Problem 1 - Setting up the project template

Description: The provided template had some minor issues when trying to run it. As Gradle couldn't diagnose the errors, we had to handpick the problems and solve them, which proved to take quite a lot of time due to the big number of files.

Solution: Rafal and Stan have looked together throughout the template and figured out that the name of the package used by Gradle didn't match the one incorporating all folders.

Problem 2 - Deciding the architecture

Description: Due to the fact that the requirements changed after we finalized our initial plans, we had to do a lot of restructuring which delayed our start of implementation.

Solution: Had additional meetings to redo the structure and finish the architecture design so that everyone has a slight idea of how the project should look like

ADJUSTMENTS FOR NEXT SPRINT

- Finish up issues on time
- Stick to an architectural plan from the beginning

SPRINT 2 RETROSPECTIVE

PROJECT: “Rowing”

GROUP NR: 29A

Student	Task (Issue nr)	Estimated Effort	Actual Effort	Done	Notes
Alex Preda	Assignment 1 draft	3h	3h	Yes	
	#8 - Implement Event creation & editing APIs	4h	5h	Yes	Create API for Listing Events compatible with a specific user
Anca Badiu	#13 - User schedule	3h	3h	Yes	
	Assignment 1 draft	3h	3h	Yes	
Stanisław Howard	Assignment 1 draft	3h	3h	Yes	
	#6	3h	6h	Yes	For now it has been adjusted to our needs it might be changed further as we progress
Rafal	#3-add-CI	1h	1h	Yes	Some help
	Assignment 1 Draft	3h	3h	Yes	
	#5-event-class	2h	2h	Yes	
	Changed project structure	1h	3h	Yes	All the changes were merged in the same merge request s #5
Violeta Macsim	#9 - Notifications	4h	3-4h	No	Notification controller to be done
	Assignment 1 draft	3h	3h	Yes	
Kayleigh Jones	Assignment 1 draft	3h	3h	Yes	

MAIN PROBLEMS ENCOUNTERED

Problem 1 - Creating the UML diagram

Description: Creating the UML with little examples and explanations about different notations posed a big challenge. Also, we were still having debates about a couple of architectural designs, which made the decisions of how the UML should look more difficult.

Solution: We have studied more about how UML diagrams should look and what they should contain in order to translate out design ideas on paper.

ADJUSTMENTS FOR NEXT SPRINT

- Do task division at the beginning of a sprint to make sure everybody has enough time to finish

SPRINT 3 RETROSPECTIVE

PROJECT: “Rowing”

GROUP NR: 29A

Student	Task (Issue nr)	Estimated Effort	Actual Effort	Done	Notes
Alex Preda	Assignment 1	1h	1h	Yes	
	#11 - Add event listing API	3h	5h	No	Implement the Schedule created by Anca in Event and User MS
Anca Badiu	#18 - Schedule converter	4h	3h	Yes	
Stanisław Howard	#21	2h	4,5h	Yes	
	#20	3h	2h	No	Worked on it with Rafał
	Assignment 1	2h	2h	Yes	
Rafal	#20 Api Gateway	3h	8h	No	The template is added all that needs to be done is to add mapping for user and event microservice
Violeta Macsim	#19 - Enqueuing API	4h	6 -7h	Yes	Prone to have some changes
	Assignment 1	2h	2h	Yes	
Kayleigh Jones	#10 - Implement User editing APIs	3h	5h	Yes	

MAIN PROBLEMS ENCOUNTERED

Problem 1 - Changing the overall structure to fit a Gateway

Description: In order to connect all microservices, we have decided to use a “Gateway”, but with the design, at that time we would have split the logic between too many microservices.

Solution: We have done some refactoring in order to delete the usage of one microservice that would be then substituted by the gateway.

Problem 2 - Implementing the Spring Cloud Gateway

Description: For Some reason, we weren’t able to add the required dependencies to the *Gradle.build* that are required to maintain Spring Cloud Gateway. We managed to add it to a different project with Java 16 so maybe that was the problem but we wanted our project to be built using Java 11 so after a lot of work we decided to abandon the idea of using Cloud Gateway

Solution: We agreed to use RestTemplates and create controllers that will map requests manually to adequate microservices.

ADJUSTMENTS FOR NEXT SPRINT

- Communicate more about design changes

SPRINT 4 RETROSPECTIVE

PROJECT: “Rowing”

GROUP NR: 29A

Student	Task (Issue nr)	Estimated Effort	Actual Effort	Done	Notes
Alex Preda	#23 - Implement schedule into user and event	2h	3h	Yes	
	#27 - Update event listing to include gender	1-2h	2h	Yes	
Anca Badiu	#22 - Tests	3h	6h	Yes	
	#24 - Enqueuing check	4h	6h	Yes	
Stanisław Howard	20# - api-gateway	5h	16h	No	Many things were revealed to be not fully working. For instance the Timeslot implementation that was lacking
	general code refactoring 30#				
Rafal	#20 - api-gateway	3h	8h	Yes	I also had to fix a lot of issues not connected directly to this ticket to actually have working requests
Violeta Macsim	#22 - Tests	2h	1h	Yes	Had to do only part of issue
	Assignment 2	3h	1h	Yes	
	#9 - Notifications	2h	5h	Yes	Continue issue #9

Kayleigh Jones	#15 - Create creator enqueue judging	8h	10h	No	Sending notifications to accepted/rejected users not added yet
	#9 - Notifications	1h	4h	Yes	Finish issue #9 and #15

MAIN PROBLEMS ENCOUNTERED

Problem 1 - Data checks

Description: When testing some APIs we have discovered that some methods don't have data checks that would see if the user has completed fields with "expected" data, without leaving any empty text fields.

Solution: We have created an issue that got assigned to one of the teammates.

Problem 2 - Tests

Description: After finishing most of the issues, we have realized that our test coverage is relatively small.

Solution: Created an issue that was split between multiple people so that we can cover as much as possible in the little time left.

Problem 3 - Errors in code

Description: Some classes contain errors that when connecting the microservices the errors would reveal themselves

Solution: Put more effort into evaluating each MR so that in the future we can save more time.

ADJUSTMENTS FOR NEXT SPRINT

(To be decided the next sprint when receiving new work)