

Sprint Retrospective - Iteration #3

Group 58

Week 3

Microservice	User Story (from User Stories doc. numbering)	Task #	Task Assigned To	Esti mate d Effor t (in hrs)	Ac tua l Eff ort (in hrs)	Done (yes/ no)	Notes
Rooms Schedule	Admin/8	Create Rule class, RuleController and RuleRepository	Nikolaos	1	1	yes	The admin can change the values of rules (e.g. slot duration, buffer time)
		Create 'rules' table in the database	Nikolaos	0.5	0.5	yes	rules(idrules, name, value) each rule has a unique id, a name and a value
		Test Rule Class and RuleController	Ee Xuan	1		no	Database connection was down, because of too many conenctions so have not been able to complete this.

	Admin/6	Create 'generateRoomSlots' method in RoomSlotController	Nikolaos	1.5	2	yes	This method calls the createRoomSlots method of the RoomSlotRepository
		Test RoomSlot Controller	Ee Xuan	2		no	Database connection was down, because of too many connections so have not been able to complete this.
		Create stored procedure 'add_room_slots' (SQL script)	Nikolaos	4	4	yes	This way the large number of INSERT operations is done in the database. Saves transfer of data.
		Create the database for Room Schedule	Nikolaos	1	1	yes	Store the connection details in the application.properties file of 'room-scheduler'
	Teacher/4	Create scheduleNewLecture method in RoomSchedulerController	Nikolaos	4	4	yes	A teacher can schedule a lecture. This method returns the info of the scheduled lecture or "There are no available slots for the input given. Try again!". For k slots, we occupy (k+1) slots (one more for cleaning the room). I created an issue for 2 improvements still to be made.
		Create getSchedule method in RoomSchedulerController	Nikolaos	4	5	yes	With this method the following information is returned for each RoomSchedule entry: {roomScheduleId, startTime, endTime, lectureId, roomId, coronaCapacity}

		Test Room Schedule and all the other entities necessary	Ee Xuan	2.5	2.5	yes	Tested all the methods in all the entities.
		Test Room Schedule Controller	Ee Xuan	2		no	Database connection was down, because of too many connections so have not been able to complete this.
	Teacher/6	Create cancelLecture method in RoomSchedulerController	Nikolaos	2	2	yes	A teacher can cancel a scheduled lecture. In that case, we delete the entries from room_schedule and we update the 'occupied' attribute of the corresponding room slots to '0' (they are available again)
Rooms		Create the database for Room	Nikolaos	1	1	yes	Store the connection details in the application.properties file of 'rooms'
		Test all methods for Room entities	Ee Xuan	0.5	0.5	yes	Testing all the methods for the Room entity.
	Admin/1	Create stored procedure 'add_rooms' (SQL script)	Nikolaos	0.5	0.5	yes	This routine is used for the creation of 20 initial rooms.
		Tested all methods for RoomController	Ee Xuan	1.5	1.5	no	Tested all methods for RoomController except the count method and getAll method.

		Create 'generateRooms' method in RoomController	Nikolaos	0.5	0.5	yes	This method calls the createInitialRooms method of the RoomRepository
Authenticati on	Student/1, Teacher/1 , Admin/5	Create login method in AuthenticationController	Alexandru	3	4	yes	This method takes the net id and password of a user and returns a JSON Web Token that grants the user access to certain endpoints based on their role.
		Create modifyUser method in AuthenticationController	Alexandru	0.5	0.5	yes	
Courses		Make api endpoints for other microservices	Luca	3	3	Yes	This is done for now, there might be some other endpoints that need to be implemented later
		Make method that schedules lectures for the coming 2 weeks	Luca	3	3	Yes	This was not that hard, but required a lot of communication with the room-scheduling people
		Create createLecture method in CoursesController	Luca	1	1	Yes	After figuring out database issues, this was not that hard
		Test all methods in CoursesController	Luca	4		No	I have not yet found the time to implement tests, this will need to be done in sprint 3
Student	Student/6	Create assignStudentsUntil endpoint in StudentAssignmentCo	Alex	1	1	yes	This mostly consisted of cross-service calls. Because of this it took some time to validate whether it was working as expected.

		ntroller					
	Student/6	Create a helper method for getting all scheduled lectures	Alex	0.5	0.5	yes	This required sending a request to the room-schedule service, which made it kind of difficult to check whether it worked.
	Student/6	Create a method for getting all students ordered by their 'go to campus priority' (change the previous method that ignored 'wantsToGo')	Alex	0.5	0.5	yes	It took quite a while to figure out what the method name in the repository should look like, as it should not only sort by start time, but also only consider students that want to go to campus in the first place
	Student/6	Implement endpoints in student service with room-scheduler service	Alex	2	2	yes	This went pretty straight forward, the tricky bit was getting everything regarding the GSON library to work as required.
	Student/6	Merge the two different RoomSchedule classes into one	Michel	1	1.5	yes	Had to refactor quite some code, to get it all in one class. But it was doable.
	Student/6	Being able to handle coronacapacity for scheduling	Michel	0.5	0.5	yes	Was mostly done in the merging of the two roomschedule classes and adding an attribute for the coronaCapacity. Requesting the RoomSchedule service to provide that information.
NA	NA	Fix project gradle structure to make checkstyle and pmd work properly as well	Alex	2	5	yes	It just took a lot of time to constantly check if all the subprojects work as desired and I often lead into issues and needed to try another approach. Also fixing all the checkstyle errors that were on the development branch that weren't detected by the previous pipeline cost a while to fix.

		as the pipeline on gitlab					
--	--	------------------------------	--	--	--	--	--

Main problems encountered/Points to improve

- 1) Often throughout the week, we could not connect to the database (too many requests error)
- 2) The problem above also affected testing (for now testing happens in connection to the real database)
- 3) We need more testing for the current functionality
- 4) We need more document-type work (uploading reports for tests etc.)