Sprint Retrospective, Iteration # 1

24/11/2021 - 1/12/2021

Issue	Task	Task Assigned To	Estimated Effort per Task (in hours)	Actual Effort per Task (in hours)	Done	Notes
	UML/database schema draft	*Julie*	*2*	3	✓	
	Implement Course services, controllers and repositories (except Lecturer database)	*Ali*	*2*	*3*	✓	
	Creating UML Diagram	*Martin*	*2*	*3*	✓	*We have discussed this together in several physical and online meetings.*
	Creating Courses and Hiring database schema	*Winstijn*	*2*	*2*	✓	
	Creating TA database schema	*Maurits*	*1*	*1*	✓	
	Uploaded GitLab issues	*Maurits*	*1*	*1*	✓	*Read the problem section.*
	Implement repository structure (Gradle	*Martin*	*2*	*2*	✓	*This was implemented during our sprint meeting. The team discussed what structure was going to be used.*

	modules)					
	Implement authentication support	*Martin*	*6*	*6*	✓	*This includes implementing the entire authentication microservice as well as adding full authentication components that can be used by other microservices to verify tokens*
	Assignment 1 Part 1	*Winstijn, Maurits*	*8*	*2*	*No*	*It is not finished yet.*
	Assignment 1, Part 2 - Design patterns	*Ali, Julie, Mattheo*	*8*	*6*	*No*	*It is not finished yet, will probably take more hours than estimated.*
	Improve upon Courses database schema	*Ali, Winstijn*	*2*	*2*	✓	

Project: TA-contract service
Group: 13B
Scrum Master: Winstijn

Main Problems Encountered

Problem 1: State of our Gitlab

Description:
George gave us the insight that the commits in Gitlab are mostly made by Martin and that it is not clear that others have done work. He also noted that we do not use the time tracking in our Gitlab.

Reaction:

The first few weeks were mostly composed of working on the draft architecture and figuring out the structure of our project along with which design patterns to use, therefore there weren't many coding tasks. This led to few to no commits on most team members' sides. Next week we will make sure the commits are evenly distributed and that our issues are time tracked inside Gitlab.

Problem 2: We need a dev branch

Description:
George told us that we should use a dev branch when working with scrum. All features are implemented on their own feature branches and those should be merged to dev. Then at the end of the sprint we merge the dev branch into main to clearly show what features got finished this sprint.

Reaction:

We understand why having a dev branch and merging it to main at the end of the sprint makes the workflow more structured. In sprints 2 and later, we will use a dev branch to make it clear which features were implemented during that sprint.

Problem 3: Miscommunication about task

Problem: There was an internal miscommunication regarding the creation of issues on GitLab. The person responsible for uploading the issues was not aware of having this task assigned.

Reaction: In sprint 2 and later, all team members will check GitLab periodically to make sure they are aware of their tasks. Furthermore, everyone will keep track of what others are doing, and inquire about their tasks throughout the sprint.

Problem 4: Documentation of our internal meetings

Description:
We have had several meetings this sprint (on monday and wednesday) and we do not have any documentation of it.

Reaction:
A note directory has been created where from now on all sessions will be documented.

Adjustments for the next Sprint Plan

Motivate any adjustments that will be made for the next Sprint Plan.

Action Points

Everyone should check Gitlab regularly.
Create the dev branch which will be merged into main at the end of every sprint.
Add all our notes to Gitlab
Add time estimations to all our issues on Gitlab.