Sprint Retrospective, Iteration # 3

8/12/2021 - 15/12/2021

Issue	Task	Task Assigned To	Estimated Effort per Task (in hours)	Actual Effort per Task (in hours)	Done	Notes
#32	Expose all still to be approved hours	Winstijn	2	3	✓	
#29	Implement retrieving contracts of students	Winstijn	1	4	✓	The one hour estimate was too little. Testing took longer than expected
#40	Implement rating TA performance	Winstijn	2	2	✓	
#37	Implement withdrawing candidacy	Julie	3		✓	Took way too long, due to getting stuck and learning how to merge properly.
#39	Block declaring hours if going over contract	Maurits	1	2	✓	Our issues should be more precise.
-	Add a document going into testing	Maurits	1	1	✓	
#42	Implement submitting hours	Maurits	2	1	✓	

#55	Clean up models and rework definitions in the ta-service.	Maurits	2	1	✓	
#27	Implement response to see state of application.	Julie	2		X	See other issues
#34	Expose all still to be approved applications.	Mattheo	2	3	✓	Took longer than expected because testing is hard.
#38	Block students from applying if they have too many candidacies	Julie	2		X	Getting to the end, but way too slow. I am just doing things way too slow.
#35	Block students from candidating 3 weeks before course starts	Mattheo	1	1.5	✓	
#28	Implement the creation of courses	Ali	3	5	✓	Took longer than expected due to refactoring deprecated Date class took some time to figure out how to serialize.
#57	Endpoint to check if a user is a responsible lecturer	Ali	3	5	✓	Took longer than expected because the first implementation was changed entirely.
#36	Implement rejecting of	Martin	1	1	✓	

	application by lecturer					
#54	*Create framework for inter-service communication*	*Martin*	*2.5*	*3*	✓	*Took less time than expected since making such requests in Spring turned out to be quite easy.*
#31	*Implement approving application by lecturer*	*Martin*	*3*	*3*	✓	
#41	*Block lecturer from hiring too many TAs*	*Martin*	*3*	*2.5*	✓	*Took less time thanks to the framework for inter-service communication.*

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

Main Problems Encountered

Problem 1: MRs aren't opened on time

Description:
This time our MR's did all exist before our deadline but they weren't yet all marked as ready. The deadline for this was Tuesday 8pm but it was missed. All MR's should be marked as ready by 8pm, and actually be ready, because otherwise it will snowball into our other deadlines on wednesday.

Reaction:
Our deadlines will be made harder. If it is not marked ready by Tuesday 8pm it will be pushed back to the next sprint.

Problem 2: Time tracking on MRs

Description:
Time tracking on MRs should be used to estimate the time it takes to review the code. The assignee of the issue should estimate this and add to the time tracker of the corresponding MR.

Reaction:
Time tracking on both the issues and MRs will be taken more seriously. After each coding session, the time difference between the first commit and the last will be added to the issue. The assignee will estimate the time it takes to review code depending on how much time they spent writing it and add it to the time tracking of the MR.

Problem 3: Julie's time management.

Description:

We have one member in the team who is generally weak compared to the other team members. Although she has put in a lot of effort most of it did not manage to get merged on time.

Reaction:

Currently we expect this issue to resolve itself up to a certain extent. It took Julie a long time to get up to speed and understand what we are doing. She still has enough issues to get her contribution back up to where it should be and we will be keeping close watch on the git inspector results to check her efforts.

Problem 4: Work should be pushed.

Description:

There were teammates who didn't immediately push all their work to get but would wait till a very late time. This stresses out project management since there is no clear view on what is happening.

Reaction:

From now all work should be pushed within 2 hours after you're done with it. Merge conflicts should still be resolved locally and `gradle build` should pass in the same way as it did before the merge before pushing. This relates to if it passed with warnings before it is allowed to pass with warnings after.

Adjustments for the next Sprint Plan

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

Action points
- Honour the deadlines set upon us.
- MRs should be opened as soon as you start working on your issue (drafts)
- Push your work to your branch whenever possible.
- Add estimated hours to review code as the assignee of the issue
- Julie finishes all open issues from sprints 2 and 3.