

Sprint Retrospective, Iteration #5, Group: 2

User Story	Task	Task assigned to	Estimated Effort per Task (in hours)	Actual Effort per Task (in hours)	Done (yes/no)	Notes
As a developer, I want to improve the quality of our code	Ask the TA to obtain peer feedback of our code	Tim	1	1	yes	
	Test the enhancements	Tim, Rens, Dennis	3	0	no	There was not much need to add tests as we did not build new features. We only improved the quality of the existing code.
	Implement the enhancements that the other group proposed	Tim, Rens, Dennis	15	11	yes	
As a developer, I want to detect code flaws with inCode and resolve those flaws	Run inCode on the code base and upload analysis report to github	Dennis	1	1	yes	
	Write in the explanation PDF file where analysis is located	Erik, Dennis	6	1	yes	
	Pick the first three design flaws in order of severity	Erik, Dennis	4	3	yes	
	Explain what design choices or errors lead to the design flaw	Erik, Dennis	8	7	?	
	Fix the flaw or explain extensively why it is not an error	Erik, Dennis	15	16	?	
As a developer, I want to extend my game to implement more features according to what our supervisor wants	Talk to the TA to obtain a new requirement for the game	Gijs	0.5	0.5	yes	
	Write a requirements document for the extension given by the TA	Gijs, Rens	2	1	yes	
	Send the requirements document to the TA for evaluation	Gijs, Rens	0.5	0.5	yes	
	Write tests for the requirements	Gijs, Rens	4	2	yes	
	Implement the requirements	Gijs, Rens	8	12	yes	
	Design a UML for the extension	Gijs, Rens	2	0.5	yes	

Sprint Retrospective, Iteration #5, Group: 2

As a user, I want to be able to play the game without bugs	Fix all findBugs errors/warnings	Rens	2	1	yes	
	Let the build fail when the above requirements (findbugs) are not met	Gijs en Rens	1	1	yes	
	Write JUnit tests for at least 80% coverage (excluding GUI)	Erik, Gijs	15			

Main Problems Encountered

Problem #1: Merges which touch a lot of lines

Description: This week we saw two merges touching a lot of lines (500+). It is really hard to review these commits and may result quickly in a messy codebase.

Reaction: New pull requests should not touch more than 200 lines. For one large feature, like multiplayer, it may be needed to separate this into multiple pull requests.

Problem #2: Overall test source quality

Description: The quality of the tests seem to be declining quickly over time from the moment we removed the checkstyle checks for the main/test folder in maven. Too little javadoc, duplicated code and inconsistency in @Before/@After calls were some of the things we heard from the review.

Reaction: We have been improving the code quality for test sources, but to keep it high quality long-time we should re-enable the checkstyle checks.

Adjustments for the next Sprint

- Re-add Checkstyle for tests.
- Do not open pull requests which touch more than 200 lines.
 - Split them up in separate features/fixes/pull requests.