Sprint Retrospective, Iteration #6

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	Enable Checkstyle for test sources.	Tim	1	3	yes	Maven-Checkstyle documentation was very unclear on this, had to build a custom way around the problem.
	Fix all current Checkstyle issues for test sources.	Tim	4	2	yes	
	Decrease length of big classes	Gijs	8	5	yes	
As a user, I want to be able to play the game without bugs	Increase branch coverage of nl.tudelft.sem.group2.collisions package to 80%	Tim	3	0	yes	Appeared to have 80% branch coverage already. Bad planning...
	Increase branch coverage of nl.tudelft.sem.group2.gameController package to 80%	Gijs	7	7	yes	
	Increase branch coverage of nl.tudelft.sem.group2.level package to 80%	Gijs	5	5	no	Reached 75% branch coverage. Was hard to test more than this as that would included GUI tests.
	Increase branch coverage of nl.tudelft.sem.group2.powerups package to 80%	Dennis	2	2	yes	

	Increase branch coverage of nl.tudelft.sem.group2.scenes package to 80%	Gijs	4	4	no	Included a lot of GUI code which was not properly testable.
	Increase branch coverage of classes in general package nl.tudelft.sem.group2 package (such as AreaTracker) to 80%	Rens	4	4	yes	
	Increase branch coverage of nl.tudelft.sem.group2.units package to 80%	Dennis	10	10	yes	

Group: 02

Main Problems Encountered

Problem #1: Hours distribution

Description: The amount of hours that some group members spent on the project was much more than that of others.
Reaction: This is probably due to the fact that we wrote the sprint backlog too late, after which some group members already started working on extending/rewriting/adding tests.The sprint backlog should be the highest priority in these kind of projects.

Problem #2: Cobertura test coverage output

Description: The cobertura test coverage output was not consistent over all machines, while being on the same branch and revision.
Reaction: This was due to the fact that for some reason cobertura may count commented lines as ‘uncovered’ on some machines. Moreover, some tests had specific configuration for Mac OS devices and therefore had different output on different operating systems.

Problem #3: 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.

Adjustments for the next Sprint

- N/A