Reflection on Sprint #4

Game: Fish.io
Group: 26

User Story #	Task #	Task assigned To	Priority	Estimated Effort Per Task	Actual Effort per Task (in hours)	Done	Notes
User Story 1 Software metrics and fixes	1 Create inCode analysis file	Luka	A	1	1	Yes	
	2 Find and explain cause of design flaw 1: YAML settings loader	Luka	C	1.5	1.5	Yes	
	3 Find and explain cause of design flaw 2: Bounding Box	Jesse	C	2	2	Yes	
	4 Find and explain cause of design flaw 3: PuFreeze & PuSuperSpeed	Luka	C	0.5	1.5	Yes	
	5 Fix or explain the existence of design flaw 1: YAML settings loader	Luka	B	1.5	1	Yes	
	6 Fix or explain the existence of design flaw 2: Bounding Box	Jesse	B	4	2.5	Yes	
	7 Fix or explain the existence of design flaw 3: PuFreeze & PuSuperSpeed	Luka	B	0.5	0.5	Yes	
User Story 2 Multiplayer	1 Requirements Document	Robin	A	2	3	Yes	
	2 Responsibility driven design document/cards	Chiel	C	2	1.5	Yes	
	3 UML	Chiel	C	2	2	Yes	

	4 Multiplayer communication to client	Taico	A	6	6	Yes	
	5 Multiplayer communication to server	Taico	A	6	5	Yes	
	6 Other multiplayer implementation parts	Taico	A	6	12	Yes	
Other	1 Write sprint reflection	Robin	B	2	3	Yes	
	2 Write sprint plan	Jesse	B	2	2	Yes	
	3 New game logo	Robin	C	2	4	Yes	
	4 Fix screen resizing	Chiel	C	7	7	Yes	
Introduced sprint items	1 GUI of multiplayer screen	Robin	A	2	2	Yes	

Every week, each group member also reviews pull requests of other members. Since we put a lot of time into this we also want to document time spent reviewing a separate table. Note that we spent similar amount of time of the reviewing of pull requests from day 1 of this project.

Group Member name	Time Spent Reviewing and creating Pull Requests (in hours)
Taico	2
Jesse	4
Luka	8
Chiel	1
Robin	4

Main Problems Encountered

Problem

Description: Time scheduling and organisational issues. In sprint four the assignment was put online on Monday. This meant that we could only start working on the assignment after Monday evening. Normally, our team always had a group meeting on Monday to do a sprint reflection and preparation

meeting for the coming sprint. This was not possible. Furthermore, we had to implement quite an extensive feature, multiplayer, in this sprint in our TA assignment. We got this assignment on halfway through Tuesday. This meant that we had 3.5 days to implement this feature and add to that, only one of our group members had sufficient experience with sockets (and a library that simplifies the creation process). Lastly, the feature was not easy to split up as each component has to work together in a server client application.

We also like to add that we have deadlines for other courses that week. This meant that we could not spend all the time on this course in our free time.

Reaction: Even though the assignment was posted too late, we already had the sprint reflection done and some refactoring done during the weekend and Monday. Doing these sprint items lessened the work load for the rest of the week a little. The multiplayer feature was implement mostly by one person, but this meant that the rest of the team took care of the other assignments and all the documentation necessary. We managed to complete a very simple multiplayer implementation that worked and got all the assignments done on time. This sprint we greatly favoured Erik Meijers saying: “Done is better than perfect.” But this does mean that we introduced some technical debt in our system in order to meet the deadline.

Adjustments for the next Sprint Plan

In the next (perhaps even the last two sprints) we will have to polish up our game and make sprint items specifically for refactoring. Also our test coverage went down a lot because multiplayer code is hard to test and we have difficulty with Travis not passing our tests while they do pass on our systems. This due to RAM and CPU requirements for launching and running our game. We have to add a lot of new tests to make up for this missing coverage and keep Travis happy.