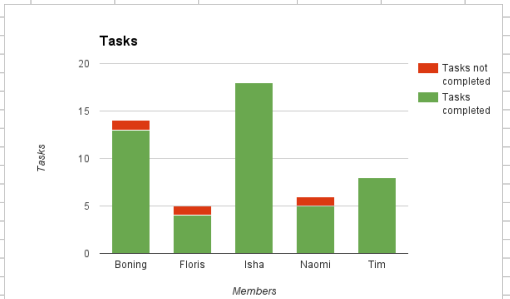
Game: Bubble Trouble		Sprint 2	Week 4											
Group: 32														
User story	**Task**	**Assigned to**	**Estimated Effort**	**Priority (1-5)**	**Actual Effort (h**	**Done (yes / no)**	**Notes**	**Points**			Sprint tasks	Dinsdag Meeting	Totaal	
As a player, I want to start a new survival game	Create a start button for survival mode	Tim	low	5	0,2	Yes	Needed some help with button backend			Boning	7,66	4	11,66	
As a player, when I'm playing a survival game, I want the game to keep getting harder and harder	After a certain amount of time, new bubbles are spawned into the game	Boning	Medium	5	2	Yes				Floris	6,7	4	10,7	
	Spawn time bubbles decreases over time	Boning		4		Yes				Isha	7,02	4	11,02	
	Bubble size increases over time	Boning		4		Yes				Naomi	3,78	4	7,78	
	Bubble spawn at random places each time	Boning		3		Yes				Tim	12,22	4	16,22	
	The player has one life	Boning	low	5	0,08	Yes							57,38	
	There is no life powerup in the survival mode	Boning	low	4	0,15	Yes								
	A new powerup can drop: score doubler	Boning	low-medium	1	0	No	Not that important, so didn't implement it							
	The score doubler gives a 2x score for 10 seconds							30			Tasks complete	Tasks not complet	Time per task	
	The game ends if and only if the player dies.	Boning	low	5	0,25	Yes				Boning	13	1	0,5892307692	
As a player, I want to get more points if the bubble has not bounced yet	The score gained by popping a bubble is multplied by 2 if the bubble has not bounced yet	Floris	medium	1	0,2	Yes				Floris	4	1	1,675	
As a player, when I finish a survival game, I want to see a leaderboard with all the highscores	When the game ends, the player can fill in his name and his name + his score is saved	Boning	medium	1	3	Yes				Isha	18	0	0,39	
	The highscores are shown on the leader board after the game ends	Boning	medium			Yes				Naomi	5	1	0,756	
	Ten highscores are shown on the leader board	Boning	medium			Yes				Tim	8	0	1,5275	
	If the player has a score higher than the lowest high score, the new score will be added to the leader board and the lowest high score will be deleted.	Boning	medium			Yes								
As a developer, I want my extension to be well thought out, and documented correctly	Use responsibility driven design for survivial game	Naomi	low-medium	4	0,3	Yes		15						
	Use UML for survival game	Naomi	medium	5	0,3	Yes								
	Add Tests for added code	Floris	medium	4	2,5	Yes								
	More coverage of tests	Floris	medium	4	3	Yes								
	Add javadoc to all methods	Floris	low-medium	4	No	No								
	Restructure code	Tim	high	2	4	Yes								
	Remove all unused branches	Boning	low	3	0,08	Yes								
	Split up methods >200 lines	Tim	low-medium	4	2	Yes								
	Remove mains from Screens	Naomi	low	3	0,08	Yes								
	Edit pom file in maven	Naomi	low	5	2,5	No								
	Add requirements to trello	Tim	low	2	0,02	Yes								
	Hand in sprint reflection of last week	Isha	low	5	0,08	Yes								
	Hand in this sprint plan	Isha	low	5	0,08	Yes								
	Hand in survival requirements	Isha	low	5	0,08	Yes								
	Hand in walls requirements	Isha	low	5	0,08	Yes								
As a developer, I want my code to be as well organized as possible	refactor levelcreator + let NormalLevelCreator and SurvivalLevelCreator extend from it	Tim	medium	5	0,5	Yes								
	refactor game + let NormalGame and SurvivalGame extend from it	Tim	medium	5	1	Yes								
	rename driver to gameScreen + let NormalGameScreen and SurvivalGameScreen extend from it	Tim	high	5	4	Yes								
	refactoring Level Class	Boning	medium	5	2									
	make a score class and refactor the code depending on it	Tim	medium	5	0,5	Yes								
As a player, when I play the game, I want walls in my level where the ball bounces against and the player can't go through	Create a wall class	Isha	medium	5	0,2	Yes								
	Walls can be added to a level	Isha	low	5	0,1	Yes								
	There are 3 subclasses of walls: Player walls, Bubble walls, and Duo walls	Isha	medium	5	0,2	Yes								
	When a bubble hits a player wall, it will go through it	Isha	medium	5	0,1	Yes								
	When a player hits a player wall or a duo wall, the player can't go further into that direction	Isha	medium	5	0,5	Yes								
	When a bubble hits a bubble wall or a duo wall, it will bounce back to the other direction	Isha	medium	5	0,5	Yes		30						
	When a player hits a bubble wall, he will go through it	Isha	medium	5	0,2	Yes								
	Neither a player nor a bubble can go through a duo wall	Isha	medium	5	0,2	Yes								
	When a certain condition is met, the wall will dissapear	Isha	medium	5	0,5	Yes								
	A condition for a wall to disappear is depending on the level.	Isha	medium	5	1	Yes								
	Example conditions are: all bubbles are popped, a timer, switch on the roof	Isha	medium-high	4	1	Yes								
	The 3 different types of walls have different colors	Isha	low	4	0,1	Yes								
	Use UML for survival game	Naomi	medium	5	0,3	Yes		15						
	Use responsibility driven design for survivial game	Naomi	medium	5	0,3	Yes								
	Add Commit tag	Isha	low	5	0,1	Yes								
	Check Commit tag	Boning	low	5	0,1	Yes	Checked							
	Create sprint plan and requirements	Isha	medium	5	2	Yes								
	Restructure drivers. Restructure the main methods. Only have one main method.	Floris	medium	4	1	Yes	Created a mainRunner and made Driver an abstract class and two implementations of this abstract class, namely: NormalDriver and SurvivalDriver. The startscreen creates a Driver according to the factory pattern							



Main Problems Encountered	
Problem 1	Too much chaos before the evening of the deadline
Description:	Lots of things went wrong in the evening before the deadline. So we had to stress to fix everything and to hand it in on time.
Reaction:	This time we will try to finish everything a day before the deadline so we have enough time to check everything before we hand it in.
Problem 2	Merge problem with walls

Description:	When we tested the walls on the different branch, they worked. And when we merged the master in the branch, they worked. But after we added more features, they walls suddenly didn't work anymore	
Reaction:	Always run regression tests when adding new features and better code reviews	
Problem 3	Two drivers with two mains	
Description:	We had two drivers in our code which had two main methods.	
Reaction:	We implemented a new class called MainRunner which was supposed to solve the issue with having two main methods in each driver. We also made a abstract class called Driver to give a good template for the other two drivers. This enabled us to make a factory and let the startScreen create the correct driver. This fix enabled us to have only one main method and much nicer and cleaner driver methods	
Problem 4	Not enough working together face to face	
Description:	We wanted to plan an extra project session together, but we couldn't find a time where every member was available.	
Reaction:	This time we will definitely find an extra day to work on the project together. This is more effective and communication is easier.	
Adjustments for the next Sprint Plan		
The next sprint we are definitely going to hand in our code sooner. Right now it was way too chaotic at the last moment, so we couldn't implement all the tasks we wanted. We also want to focus more on testing, and making sure the test work even after we add new features. We will do this by regression testing after every commit. For the next sprint we're also gonna add who is working on a task, in addition to who is responsible for a task, so we can track how many hours each group member spend on the project. Next time we will do better and more code reviews for higher code quality.		
https://blackboard.tudelft.nl/bbcswebdav/pid-2564951-dt-content-rid-8810024_2/courses/36658-151601/sprintReflection.pdf		