Sprint Retrospective, Iteration #1 - End of the week update

User Story	Task	Task description	Member responsible for the task	Task Assigned to	Estimated Effort per Task (in hours)	Actual Effort per Task (in hours)	Done (yes/no)	Priority A - E (A is highest, E is lowest)	Note(s)
Track *(will be elaborated later on)*	Track improvement Generalization Pixel based		Joost		4 hours	0 hours	Not done	E E	Since our experience with both Java and responsibility driven design is increasing by the week we are considering to start over 'fresh' after we have decided on the number and type of classes and have produced the class diagrams. This would involve setting up a new GitHub repository. During the week we decided to focus more on project management than code writing, which is why (virtually) no hours have been spent on the initially planned tasks, but most hours were spent on preparing for Thursday's meeting, the actual CRC card session and some on the UML diagrams. Requirement Traceability has been started up as well. For every 'change request', a new matrix will be generated, to keep track of changes in the project.
Shooter *(will be elaborated later on)*	Rotation Accuracy improvement		Callum		4 hours	0 hours	Not done	E	
Marbles *(will be elaborated later on)*	Marble position update New class???		Matt Xinyue		4 hours 4 hours	0 hours 0 hours	Not done	E E E E E	
Intersection *(will be elaborated later on)*	Intersection		Development of the intersection function		4 hours	0 hours	Not done	E E	
CRC card session	CRD card session Preparation of session		Team Rodrigo&Matt		1 hour 2 hours	1 hour 2 hours	Done	A A	
Exercise 1	UML Class diagram UML Sequence diagram	After deciding in general what classes we expect to need for the game and their interactions, UML Class & Sequence diagrams will be made. The task is divided in two, the main, gameplay and GUI classes being the first and the rest of them being the other	Xinyue Rodrigo	Xinyue Rodrigo	1 hour 1 hour	2 hours 2 hours	Work in progress	A	
Exercise 4	Maintaining Traceability	A strategy for keeping and maintaining traceability is being chosen. A traceability matrix will be used to describe the presence of a link between artifacts.	Matt Joost	Matt Joost	1 hour 1 hour	4 hours 2 hours	First versions of Req. Trac. Matrices	A	

Project: Zuma Deluxe
Group: Deluxe

Main Problems Encountered

Problem 1

Description:
We all agreed on the fact our code doesn't fit to be used in a later phase of the project.

Reaction:
Use of CRC cards and UML we want to figure out a proper way to build up the structure.

Problem 2

Description:
All classes were built without a real plan of execution so the classes weren't able to interact with each other after they were finished.

Reaction:
Use of UML to display what each class needs as input and can output so the merging wouldn't be that hard in the end.

Adjustments for the next Sprint Plan

For now leaving the code of the initial version as it is and (res)start by using the CRC cards, UML diagrams and the other exercises to start building up a fresh base.