Reflection on Sprint #3

Game: 2048
Group: 21

| User Story # | Task # | Task Assigned To | Estimated Effort | Actual Effort | Done | Notes |

| Assignment 1 | 1.1 | Paul & Arthur | Easy | Easy | Yes | We used CRC cards | | | 1.2 | Paul & Arthur | Easy | Easy | Yes | We changed and merged a lot of classes | | | 1.3 | Paul & Arthur | Easy | Medium | Yes | We used UMLet to make the UML | | | 1.4 | Paul & Arthur | Easy | Medium | Yes | We used UMLet to make the sequence diagram | | Assignment 2 | 2.1 | Arthur | Easy | Easy | Yes | None. | | | 2.2 | Arthur | Easy | Easy | Yes | None. | | | 2.3 | Paul & Arthur | Difficult | Difficult | Yes | There were no cases where we could apply double dispatch.| | User Story 1 | 1.1 | Jente | Medium | Difficult | Yes | The actual implementation was simple, although before we could do this we had to re-write the entire codebase. | | | 1.2 | Jente | Medium | Difficult | Yes | The actual implementation was simple, although before we could do this we had to re-write the entire codebase. | | User Story 2 | 2.1 | Jochem | Medium | Medium | Yes | None. | | | 2.2 | Jochem | Medium | Medium | Yes | None. | | | 2.3 | Piet & Jente | Difficult | Medium | Yes | After the rewrite, we only had to figure out how to manage our screens. | | | 2.4 | Piet & Jente | Difficult | Easy | Yes | We only had to launch a new MenuScreen. | | | 2.5 | Piet & Jente | Difficult | Medium | Yes | It was just a case of polling for a connection and then launching the MultiGameScreen. | | User Story 3 | 3.1 | Jochem | Medium | Medium | Yes | Player does not return to menu immediately, but has to press a button instead. | | | 3.2 | Jochem | Medium | Medium | Yes | See note for 3.1 | | | 3.3 | Jochem | Medium | Difficult | Yes | See note for 3.1 | | User Story 4 | 4.1 | Piet & Jente | Difficult | Medium | Yes | The foundation was already there thanks to Jochem and our ScreenHandler. |

Main Problems encountered

Problem 1

Description: version 1.0.0 was not suitable for extension, as we did a lot of things ourselves without using LibGDX' classes. This would result in a huge and unmaintainable codebase, if we were going to make use of menus and buttons and whatnot. As such, we decided a new setup was necessary. We found out that LibGDX has some very nice classes to help us out (Stage, Actor, Group) but to make use of these, we had to rewrite our entire codebase.

We underestimated this, so it took a lot of our time and it came with a fair share of issues to solve.

Reaction: we worked our asses of to get everything done in time. Next time however, we will do more research before starting so we can avoid such

rewrites altogether.

Problem 2

Description: testing was a huge issue this sprint, because our gameobjects now extended the Group and Actor classes. This means that they require textures, that aren't available during headless testing.

Reaction: our solution was to add second constructors to classes requiring this, so we could insert mock objects upon creation.

Adjustments for the next Sprint Plan

For the next sprint plan, we will divide the difficult tasks more between the group members to split the workload more evenly. Besides this, we were happy with the previous sprint.