COMP20050 Scrum Notes

Team: MVP

Assignment: 2

ScrumMaster: Edwin

**Sprint Backlog**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Task** | **Owner** | **Estimate** | **Due** | **Done?** |
| Incorporate blank tile into check for word | Gerard | 2 | 12/2 | Done |
| Check for move | Joe | 2 | 12/2 | Done |
| Create ascii board | joe | 2 | 12/2 | Done |
| Place (word score) multipiers | Gerard | 1 | 12/2 | Done |
| Change square to have null tiles not letters | Joe | 2 | 12/2 | Done |
| fix toString for board | Joe | 2 | 16/2 | Done |
| placeWord on board | Ed | 2 | 16/2 | Done |
| Reset method for board | Ed | 2 | 16/2 | Done |
| Nullify (word score) multipliers | Ed | 1 | 16/2 | Done |
| Complete boards toString | Gerard | 2 | 16/2 | Done |
| Add adjactent conditions to check for move | Ed | 2 | 18/2 | Done |
| BoardTests to be built | Gerard | 3 | 18/2 | Done |
| Stop ability to overlay words | Joe | 2 | 18/2 | Done |
| Create/Fill Constants Class | joe | 1 | 18/2 | Done |
|  |  |  |  |  |
| Pull notes together for assmnt submision | Ed | 1 | 20/2 | Done |

**Test Plan**

For manual test only

|  |  |  |
| --- | --- | --- |
| **Feature** | **Owner** | **Pass?** |
| Check a sample square for score value | JOE | Y |
| Play a word in a location thats out of bounds | Gerard | Y |
| Play a word in a location that doesn't connect | Gerard | Y |
| Play a word that isn't in the frame | Gerard | Y |
| Play a word in the Start location | Gerard | Y |
| Check that a played word is infact there | JOE | Y |
| Play an Adjacent Word | Gerard | Y |
| Play an Intersecting Word | Gerard | Y |
| Play an extention to a previous word | Gerard | Y |
| Ensure correct score is incremented | Gerard | Y |
| Play an Overlapping word | Gerard | Y |
| Print the board | JOE | Y |
| Reset The Board | JOE | Y |

**Review & Retrospective Summary**

|  |  |  |
| --- | --- | --- |
| **Problem** | **Resolution** | **Lesson** |
| Passing letters to the board did not allow for accurate score keeping | Passed whole tile, instead of just letter, to square. | Reviewing how information within the game is dealt with improved how we handled it. |
| Scores were being calculated incorrectly | Multiplier needed to be nullified after it had been used. | Forward planning and anticipating problems can help save time later on. |
| Frame does not get re-filled after playing a word | Frames checkForWord method was ammended | Some times you need to amend old code to make it comply with new code. |
| Multiplier logic was flawed | After a discussion we implemented a sound argument to deal with word and letter mulitpliers | Team conversations can solve problems |
| the \* was effecting how words were being played even after the first word | Needed to adjust the logic involved in placing words | Again, team work in looking at problems helped give insight to fiix. |
| Errors where words were being placed | Over complicated algorithms needed to be simplified. | Sometimes the most straighforward way is the best. |
| Many different areas needed to be adjusted as the program becomes more complex | Without a clear definition between classes, a much more in depth section was created for assigning tasks. | Communication is very important through the group |
| Constant values are everywhere | A Constants Class was made | Keeping constant declarations together makes for cleaner code |

**Class Diagrams –** *Please see Ass2ClassDiagram.png and ProjectClassDiagram.png*