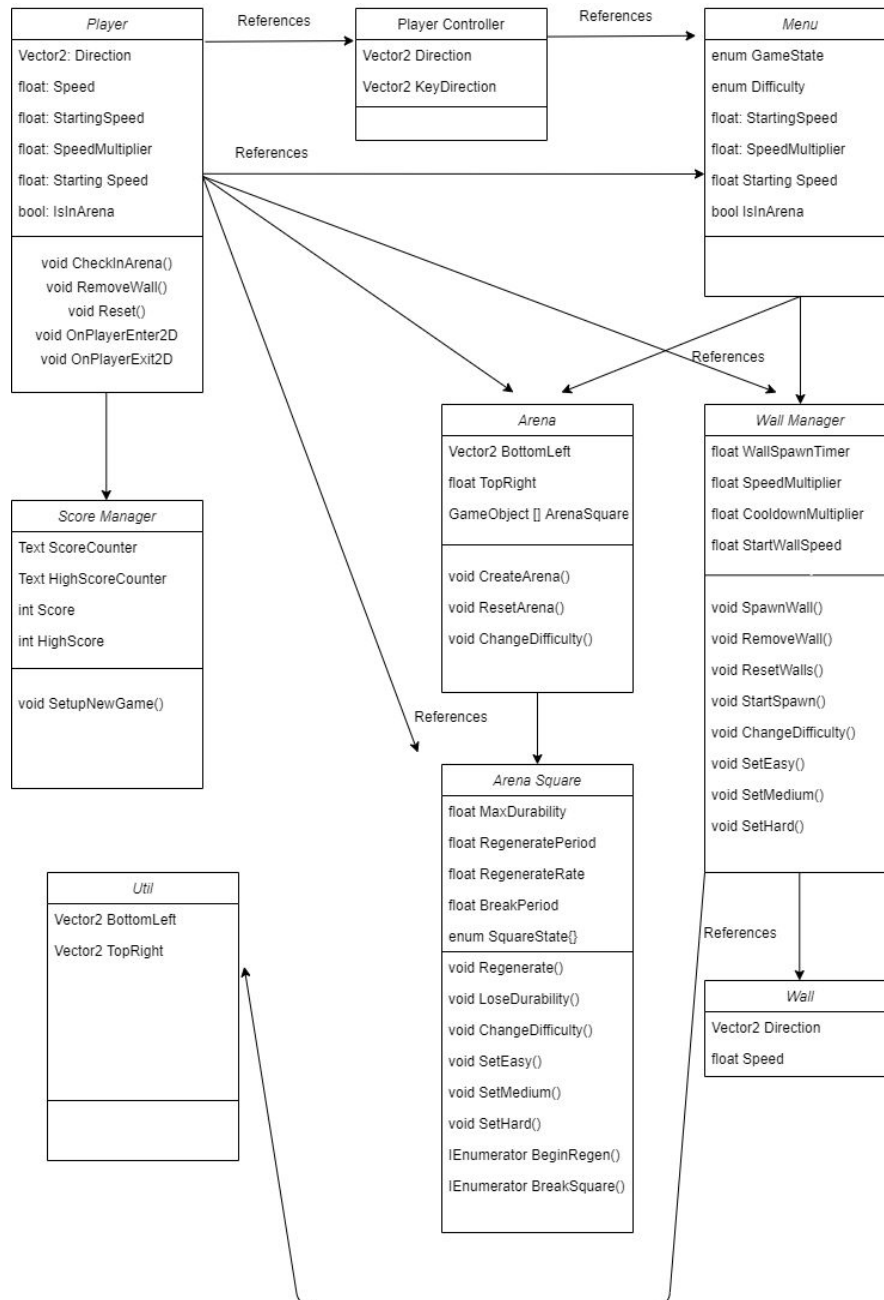Ian Jeannin
12/14/2020

Hole in the Wall Postmortem

UML Diagram:

Postmortem:

In terms of dependencies, there are quite a few. The Player class especially requires numerous other classes to function. The addition of a Menu script contained within the same scene also proved to be a poor decision in terms of the handling of dependencies, as it required reworking other scripts to be more dependent than they were originally, in part due to containing the difficulty variables that should be refactored into their own script. If I could start over, this would be the first thing I would keep in mind and want to change.

However, the Arena, ArenaSquares and ScoreManager scripts are all mostly self-contained and could be reused. WallManager originally also was, but the difficulty system caused it to be dependent on the Menu. The game is also fairly specific to Unity, as many of the scripts use GameObjects and prefabs chosen in the editor.

The game is relatively maintainable, with more walls easy to add using prefabs, and the arena could be split into more squares. If I wanted to access individual squares to create alternate maps, it would require the creation of an additional method but wouldn't be too complicated. Difficulties are also easy to add, though not as easy as they would be had I created an individual Difficulty class.

I consider the game to be finished, as it has the features I wanted complete. Most of the further changes I could make would be refining the code. I believe I chose a well-scoped out project, as I managed to finish everything but didn't have a ton of extra time to spend on it. If I could change anything in the proposal, it would simply be to make it more, as I wasn't originally sure of how I wanted the mechanics (especially the arena squares) to work.