

Code Readability Review:

Author of the code review: Nick Rieder

Date of the code review: 2/14/22

Sprint number: 2

Name of the .cs file being reviewed: Command Folder

Author of the .cs file being reviewed: Zach Richter

Number of minutes taken to complete the review: 5

Specific comments on what is readable and what is not: The way Zach implemented ICommand was very succinct and, thus, easy to read. All .cs files are around 6-7 lines of code, making it very easy to read and digest.

Code Quality Review:

Author of the code review: Nick Rieder

Date of the code review: 2/14/22

Sprint number: 2

Name of the .cs file being reviewed: Command Folder

Author of the .cs file being reviewed: Zach Richter

Specific comments on code quality: The way Zach implemented ICommand is exactly how it was shown in class which means the quality for this code is great. No complaints.

A hypothetical change to make to the game related to file being reviewed and how the current implementation could or could not easily support that change: Even if two player functionality was added to the game, the only thing that would need changing with the command functions is mapping controls for the second player. ICommand itself should already support two players executing commands at the same time.