Summary of Project

Our project codes for a one player Battleship game. The player chooses the locations for their three ships (of lengths 2, 3, and 4) and places them on an 8x8 grid. The computer randomly places its own ships ensuring that they all fit on the grid. The game alternates taking turns between the human player and the computer and when a battleship is sunk it prints "You Sunk my Battleship!" The game is over when all three of one player’s ships are sunk. The computer uses the Math.rand function to place its ships and to guess a place to attack.

The computer uses the random function to guess a place to shoot. If it is a hit then it will attempt to fire in the places around it. \*\*\*\*\*NEED TO EXPLAIN HERE HOW WE CODED FOR THIS\*\*\*\*\*

We incorporated object oriented concepts such as inheritance and aggregation which are demonstrated through the UML diagram below. \*\*\*\*\*\*ATTACH UPDATED UML HERE\*\*\*\*\*\*

We used 2-dimensional arrays as a data structure to \*\*\*\*\*WHAT DID WE USE 2D ARRAYS FOR\*\*\*\*\*.

We also implemented GUI Design. \*\*\*\* EXPLAIN WHAT WE IMPLEMENTED AND HOW \*\*\*\*

The program engages in error checking through \*\*\*\*HOW DID WE DO ERROR CHECKING\*\*\*\*

Sample Code

\*\*\*\*INCLUDE SAMPLE CODE SEGMENTS WITH EXPLANATIONS OF WHAT IT DOES\*\*\*\*

Unique Features/Elements

\*\*\*\*\*EXPLAIN HOW THE SYSTEM PLAYS THE GAME\*\*\*\*\*

Distribution of Work \*\*\*\*FILL IN WHAT YOU DID\*\*\*\*

* Sam:
* Jeremy:
* Hanser:
* Maha:
* Will:

Appendix

* \*\*\*\*\* INCLUDE SAMPLE OUTPUTS \*\*\*\*\*