Joseph DiPalma, Annan Miao, and Ben Xu

May 2, 2017

Professor Brian King

CSCI 205 Final Project

User Manual

This is a simple version of the game Battleship with graphically. This allows people to play the game Battleship on a computer over the network without having physically own the Battleship board game. This is useful for kids who enjoy this game but not able to play it physically or unable to easily find people to play with them. The overall goal is provide enjoyment for people, mostly targeted toward kids.

We didn't decide this project at first since we are just brainstorming different ideas. We know that we want to do a game, and we also considering the time requirement and the complexity of the project since we only have three weeks to complete this. Then we started to look through different types of board games, and we found Battleship, which is quite a nice game to play, and it has enough complexity for three weeks while not being too complicated since we only have three people working on this. Hence, we decided to make Battleship into a game for people to play on a computer instead the physical board game

Since we are making the game Battleship, we have multiple problem to deal with. We need to have some way to represent the game graphically, which allows users to easily navigate through the interface just by looking at it. Also, we need to establish network connections properly for multiplayer. In summary, the problems that we need to deal with are the following:

1. How properly establish connections between computers?

2. How properly display the graphical version of game that is clear and concise?
3. How to ensure gameplay goes smoothly for enjoyable gaming experience?

We decided to break down the entire game into actual objects that made up the game itself since this is very easy to think about during development. We created objects to represent the actual objects for the board game Battleship. Then we just have to deal with object interactions just like the board game does, which is very easy to consider during development.

[INSERT BASIC UML DIAGRAM HERE]

Each player has a board to show the ships, as well as an array that store the ships. Each ship object is classified by its types just like the board game does and store its location as an arraylist using 2d coordinate system. The interactions of these objects are encapsulated within the graphical user interface within the View, Model, and Controller classes.

Instruction to use

Let p1, p2 be players. Either player can choose to host the game. Suppose p1 is the host. This would an 6 stage process.

1. p1 would run the application and select host game button, and tell p2 the display ip address.
2. p2 would run the application and selection connect to opponent button. p2 would enter the ip address provided by p1, and click connect.
3. Once the connection is established, the game would start off to place the ships for both player.
4. When either player finish placing the ship, click confirm ship selection.

5. Once both player have finished placing their ships, the game would start with the host, in this case p1, attack first.
6. The game ends after either player have no ships remaining, and the whole process needs to be repeated for a rematch.