**Module Development Folder**

**Team**: B

**Members:** Mike Sawtelle, Benjamin McDowd, Bryon Lunt, David Clark

**Module**: Referee Class

**Specifications:**

The referee class makes up the “brains” of the program. It contains functions that validate the placement of pieces, that the correct player is making a move, and the actual legality of the move being made. It does this by receiving an array containing the entire board and then a square that the user desires to move. First the referee class checks to make sure the player with the right turn is attempting to move a piece and then sends a list of potential squares for that piece to move to. Then the referee class receives two squares from the GUI, one with the piece desiring to be moved and the other as the desired destination. The referee class checks that both are in accordance with the list of moves that were previously sent as permissible and then sends a code back to the GUI describing how to change the board.

Coding was only done to permit basic movement of checkers and kings without the ability to jump. It also checks piece placement, excluding the placement of mines, safe zones, and blocked squares.

**Test Strategy:**

The bulk of testing for the first build was performed here. To test the movement specifications we looked at what happens during a number of different unique scenarios for both a checker and a king during both 8x8 size boards and 10x10 size boards:

* Is the placement of checkers and kings available only in legal positions?
* Does a piece give correct options when it’s near the board’s end? Can it jump off?
* When a piece is surrounded by other pieces are the moves allowed normal?
* When a piece is blocked from any movement what happens?
* If moves are retraced what happens?
* And other similar moves.

Result: the Referee class passed as sufficient.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Build Lead Signature