## BoardDisplay

### CRC

|  |  |
| --- | --- |
| BoardDisplay | |
| Responsibilities: The purpose of the BoardDisplay class is to create a visual for the Board model and display it to the user. | Collaborators: BoardDisplay requires a gameView to be active so that the board can be rendered somewhere for the user. It also requires there to be a complete Board model already as well. It inherits from the ActionView class. |

### Description / Overview

BoardDisplay is a class that simply displays the model of the Board in the view. BoardDisplay is the parent of GameView and represents one of our views in a model-view-controller architecture. The board is to draw all the elements of the game state and redraws when told to. The board state can change from the game being played or which an observer changing the fog visibility settings. While observing, they can either see the entire board, or selected teams fog of wars to help the observer see the game from their point of view.

### Instance Variables

#### isFoggyOut

Data Type:boolean

If you are observing a game, you can set this to false to show the entire board. If set to true it will display the teams point of view based on which teams are selected.

#### visibleTeams

Data Type:List<TeamColour>

A list of teams that’s perspective is visible in the fog.

### Method Overview

*public abstract void setVisibleTeams(TeamColour … colours)*

*public void setFoggyOut(bool enableFogMachine)*

*public abstract void drawBoard()*

### Method Writeups

#### public abstract void setVisibleTeams(TeamColour … colours)

Set the list of teams that are visible in the fog.

#### public abstract void drawBoard()

Draws a hexagonal board to be displayed in the GameView. It draws the entire game state; which includes teams, robots, etc. If the user is in observer mode then it will draw based on whether fog of war is selected and which teams are selected.