## GameView

### CRC

|  |  |
| --- | --- |
| GameView | |
| Responsibilities: GameView is responsible for displaying the in game screen to the user. | Collaborators: GameView collaborates with GameMaster and inherits from BoardDisplay to obtain its goal of displaying the game view to the user. |

### Description / Overview

GameView in our model-view-controller architecture is to represent a view that will display the game state to the user. The game to be displayed contains the hexagon shaped board which GameView will know how to do since it inherits from the DisplayBoard class. Apart from drawing the Board, it will also need to draw the entire game screen; this includes things such as buttons that will shoot, or move for a selected tile. It will also need to draw the end turn button, rules button, and exit match button. At the top of the view it will indicate which team’s turn it is and on the game board it will draw the current state of each robot for each team. The view exists only when a game is created and the game still exists; one a game is finished the view will be replaced by the end game screen. The view should automatically be redrawn when a change in the game has been made to display the most recent version of the model.

### Instance Variables

#### gameMaster

Data Type:GameMaster

A reference to the controller of the current game which is called the GameMaster.

### Method Overview

*public GameView(GameMaster gameMaster)*

### Method Writeups

#### public GameView(GameMaster gameMaster)

GameView is the constructor for the class the draws the initial game in a frame to be shown to the user whether they are a player or an observer. It listens events triggered via buttons and tile selection in the view; and communicates with the game controller by having a reference to it when created. The controller (gameMaster) will handle all events when fired and GameView will redraw whenever changes to the Game Model are made.