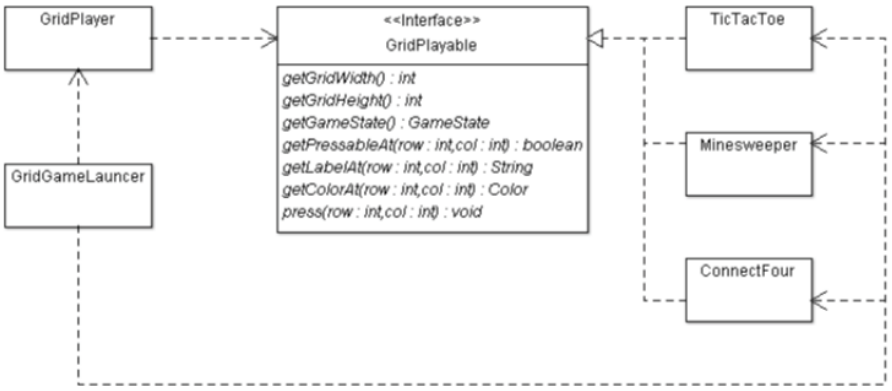
1. Problem Statement

The programmer must create a suite of games which implement the given GridPlayable interface. A separate GridGameLauncher must be able to launch a GridGamePlayer for each type of game in the suite. The GridGamePlayer must know nothing pertaining to each specific game, but must be able to interact with the GridPlayable interface.

Requirements:

1. TicTacToe Game
2. Connect Four Game
3. Minesweeper Game
4. Game Player
5. Game Launcher

2. Planning



I used the UML diagram provided to organize the layout of my program. I decided I would begin with the GridGameLauncher, because it was to be the bridge between the GridPlayer and the games. I then planned to implement the GridPlayer in order to test that the GridGameLauncher and GridPlayer were able to launch the given TicTacToe.class file. Once these three initial files worked together, I planned to implement ConnectFour.java and Minesweeper.java and make the necessary adjustments to the GridPlayer.

3. Implementation and Testing

To create the GridGameLauncher, I used an extension of JFrame to create a window that would hold three buttons. Each button, when pressed, would launch a GridPlayer for its respective game. The GridGameLauncher would also hold the main method, which would simply instantiate a launcher.

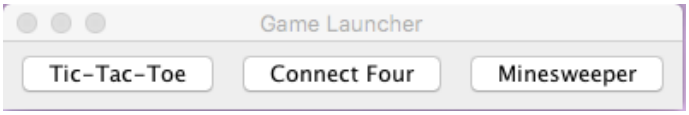
To create the GridPlayer, I used an extension of JFrame to create a window that would hold a panel of buttons. The panel would have a grid layout in order to represent the game board. During instantiation a button would be added to each space in the panel and also a two dimensional list of buttons. The GridPlayer would use a mouse listener to monitor for action in the game. The mouseListener would determine where on the grid a press happened, and then, depending on the game state, would take the proper actions and reveal the labels and colors necessary for the game. When the game state would indicate an end state, the gamePlayer would close on click, but keep the game launcher open in order to launch another game if desired.

To create ConnectFour, I used an implementation of GridPlayable in order for the game to be playable through the GridPlayer. In order to keep track of the values, I used a three dimensional array of strings to hold markers for: the label, the color, and the “pressablilty.” I also used helper methods to check for each instance of a win, a full board, and falling. The falling method was used to force the labels to appear at the lowest possible space on the board.

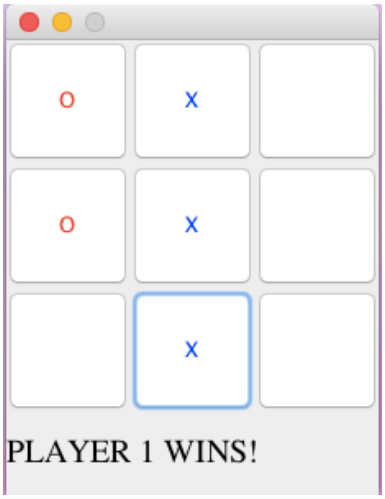
To create Minesweeper, I used an implementation of GridPlayable in order for the game to be playable via the GridPlayer. In order to keep track of the values, I used a similar three dimensional string array to hold markers for: the label, the color, "pressability," and visibility. When initializing Minesweeper, 20 bombs would be put in random spaces on the board. After this every other space would be filled in with the appropriate label according to the presence or absence of a bomb neighboring it. I used helper methods to enable the spaces to look at the spaces around them for bombs, and a pressAround method to reveal all the appropriate empty spaces.

4. Reflection

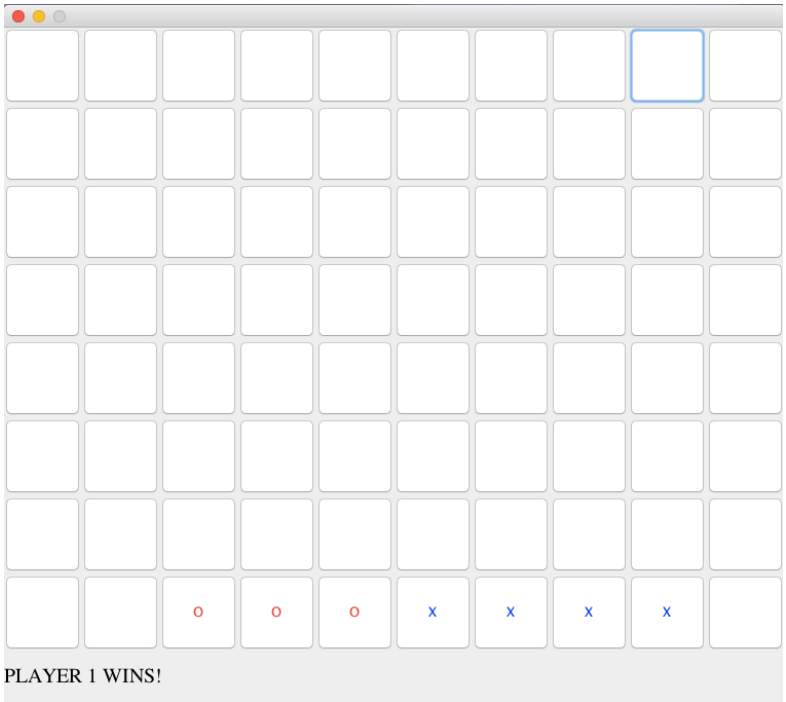
I found this project to be the most challenging of all other previous assignments. The most challenging task was the implementation of the GridPlayer. I feel this is because I was writing to fit a class file that I was not familiar with. I also found difficulty in manipulating the swing components, because of the complexity associated with them. I was unable to change the background color of the buttons in order to differentiate from previously pressed buttons and unpressed buttons. I feel the majority of my time was spent searching through the API pages for appropriate methods to suit my needs. I felt the implementation of each game was the easier task in this lab. In refactoring I would find a way to convert pressed buttons into JComponents in order to differentiate from unpressed buttons and be able to manipulate background colors and labels a bit easier. I feel like the code is overall messy and I would like to clean it up a bit also.



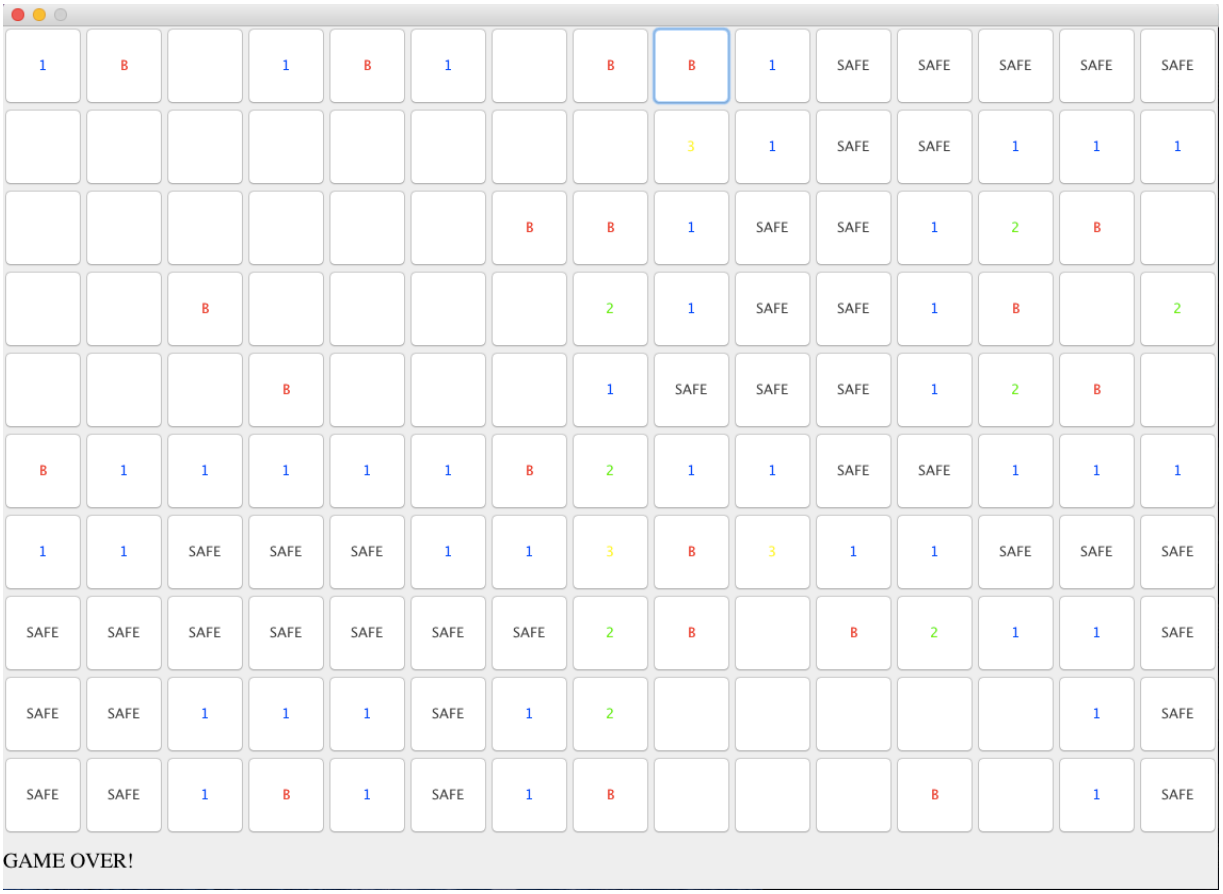
Game Launcher



Tic Tac Toe



Connect Four



Minesweeper