Assignment 4 Report

SE 319

Brad Gannon

I had an internship over the summer with Corteva Agrisciences to create several GUI components of a crop information application, which ended up being lots of programming with JavaFX. For that reason, I went very in-depth with this assignment, as my experience with user interfaces is centered around JavaFX.

I wrote more code for this assignment than was probably necessary, but here is the process I took to design this Tic-Tac-Toe game:

* First, I extended the class from the Application class, which provides several useful methods
* I set the scene, title, and then initialized the game
* For initializing the game, I created a GameSet class to basically create a scene if one did not previously exist, or else set the skin
* This GameSet class also has the quit method, if the player decides to end the game
* I created a GameControls class that extends HBox, which essentially takes care of the option to play again with buttons
* If the player chooses to play again, a newGame method is called, else the quit method is called
* I used a StatusIndicator class, again extending HBox, to show the current player information below the board
* This class also appropriately displays the icon (‘X’ or ‘O’) next to the current player text
* The Game class is the most important class, as it contains all information about the game
* It has several methods to return simple information, as well as using a switch-case statement to determine the next turn and determining the winner
* The GameSkin class extends VBox to apply the skin to the game board
* The WinningStrategy class simply implements the conditions to win or draw in this game, which was a lot of for-loops and if-statements
* If there is a winner, this class appropriately calls methods to prompt the player to play again
* The Board class contains the actual 3x3 matrix used for the game
* BoardSkin was a simple nested for-loop to apply the skin to each element of the 3x3 board
* The Square class contains an Enumeration for the three states of each element of the matrix, as well as a method to return the state of each element
* It also has a method to apply one of the given icons to a square if a mouse click occurs inside it
* Lastly, the SquareSkin class is dealing with the actual images, by importing them and creating an ImageView object that starts as being empty
* I set the dimensions and then use a switch-case statement to determine which player has the turn, then applying the image to the ImageView object and therefore into the game

For design purposes, I found it useful during my internship to create a CSS file for the design and import it into the JavaFX project. This makes the visual components (color, text size, font, etc.) much easier to set and change.

Here are screenshots of desired output:





