Jesse Barrett

10-19-21

Link to my projects: <https://jesse-barrett.github.io/N320/>

I started my project by drafting out what my Connect Four would look like manually in the HTML. Once I had decided how I wanted my SVG’s to be laid out, I hit the books and began my plan (see below). I didn’t end up sticking to my plan very well. I added an additional “Game” class and more/less properties and methods to my other classes that I hadn’t anticipated. I hit some speed bumps and bugs along the way, and there are some things I wish I did differently, but I’m proud of my work at the end of the day.

Starting at the lowest level, I have a “Chip” class contains the properties of a chip and instructions for how to draw one. Moving up, the “Board” class contains properties of the board and instructions for how to draw it. When the board is drawn, it also draws all 49 chips. There would be 42 chips, but my method for determining if a chip is valid to be chosen by a player is based on if the chip below it is taken or not. The 7 extra chips are invisible, and exist below the bottom row as “taken” so that the bottom row is a valid place for the player to place a chip. One level higher than the board is the “Game” class, which creates a new instance of “Board” when its “start” function is called. It also adds the click listeners to each chip. The “Game” class handles all active operations that can take place in the game.

Here is where I encountered my big speed bump. In my “click” function that gets called when I click on a chip, I wanted to use the “this” keyword to refer to my “Player” class and retrieve the current player. Because of the nature of the function, “this” referred to the chip that I clicked on instead of the “Game” class. My click function ended up being over 100 lines long until I could figure out the issue. I circumvented my problem by modifying my event listener as follows: *“.addEventListener(‘click’, this.click.bind(this));*”. That addition allowed me to use the keyword in the way I wanted and further split my “click” function into more specific functions like “changeChip” and “checkWin.” From there I just had to grant a win to the player (which was a short function) and then my game was complete.

Diagram

Description automatically generated