Displaying the Data:

The grid is represented using a gridPane. Each square will be a button that will only display a number if the number is not 0. The square buttons are created through a class called Square buttons which extends from button.

The constraints are displayed as labels, if there is no constraint an empty label will be printed.

Squares that are given when the user creates a new puzzle( non-editable ones) have a different ID which is styled in a different way through the CSS, making it clear for the user which ones are editable and which ones are not. The buttons from these squares are also disabled, so that there is no chances of the user accidentally clicking them.

Editing the Data:

The buttons are binded to the value of the squares, once a button is clicked it will change the value in the square, making the text in the button change as well. The number will be displayed in a red font if it violates the rules of Futoshiki. EG: if the button changes to a 3 and there´s a 3 already in that row.

The id of the buttons is set to wrong when the value is illegal, that is how the colour of the font changes.

**Extra**

Initially a main menu is displayed which lets the user choose the size and difficulty of the grid. The way the size works is by inputting a number in the textfield, if something that is not a number is inputted the textfield will just automatically delete it. If the number is not in between 5-10, the user will be shown a warning message saying that the size is out of bounds and that it has to be in between 5 and 10.

Once the game starts, if the user chooses to give up, the puzzle will display a message saying the user has lost the game. And will redirect the user to the main menu Screen allowing it to start again.