User Story 103

Description

"As a user, I want to be able to start a new game at any time so that I can play a different puzzle"

Requirements

- The user should be able to start a new game even if they have not completed their current game and should be able to do so from any game mode

Integration Test Case

- Using electron show a dialog box and if the user selects "Yes" then the function mainMenuScreen() will be called and return the user back to the main menu. If the user selects "No" or "cancel" the user will stay on their current game mode

Files

main.js

Code

```
// dialog box warning user when clicking new game label
const options = {
    type: "warning",
    buttons: ["Cancel", "Yes, please", "No, thanks"],
    defaultId: 2,
    title: "Warning",
    message: "Do you want to do this?",
    detail: "Starting a new game will delete your current game",
};
```

Figure 1: Dialog box options

```
label: "New Game",
click(item, focusedWindow){
    dialog.showMessageBox(null, options, (response) => {
        if (response == 1){
            mainMenuScreen();
            focusedWindow.close();
        }
    });
}
```

Figure 2: Dialog using electron

```
function mainMenuScreen(){
    menuWindow = new BrowserWindow({});

    menuWindow.loadURL(url.format({
        pathname: path.join(__dirname, 'menuWindow.html'),
        protocol: 'file',
        slashes: true
    }));

    const mainMenu = Menu.buildFromTemplate(mainMenuTemplate);
    Menu.setApplicationMenu(mainMenu);
}
```

Figure 3: mainMenuScreen() function

Explanation

When pressing the "New game" button a dialog box will appear and populate itself with the contents of options (**Fig 1**), and the user will be given the option between "Cancel", "Yes, please!" and "No, thanks" because these options are defined within an array we can refer to each by their index (0 = "Cancel", 1 = "Yes, please!", and 2 = "No, thanks"). When the user selects "Yes, please" the response value will equal 1 and the function mainMenuScreen() (**Fig 3**) is called. In this function we use electron to create a new browser window and load our menuWindow HTML file. After the HTML file loads the focused window being our previous game session is then closed by using the focusedWindow.close() (**Fig 1**) method provided by electron. However, if the user selects either "No, please" or "Cancel" the mainMenuScreen() method call will not occur and the user can continue on their current game session. This dialog box warning system was implemented in the event that the user misclicks the new game button and does not lose their game session on accident.