User Story 101

Description

“As a user, I want to be able to change the difficulty of the game so that I can have a variety of challenges”

Requirements

- The user should be able to choose between easy, medium and hard difficulties
- When the user chooses between an easy, medium or hard game mode the respective grid size of 3x3, 6x6, or 9x9 will be shown to the user

Unit Test Cases

- Call the function used to create the grid for each difficulty and return the value of the size of the grid and compare it with a premade gird of the correct size

Files

test.js
userStory101.js

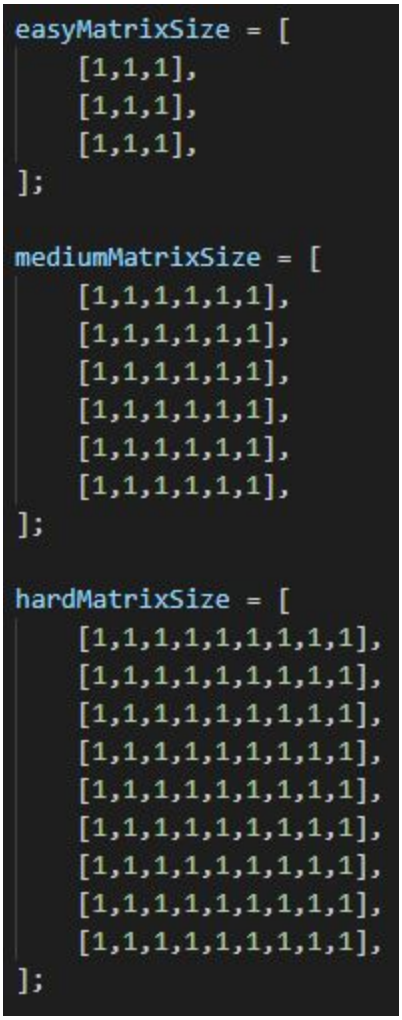
Code

```
// Unit tests to make sure the correct grid size is created for each difficulty
it ('Correctly creates the grid for easy mode', () => {
    assert.equal(operations.createGrid("easy"), easyMatrixSize.length);
})

it ('Correctly creates the grid for medium mode', () => {
    assert.equal(operations.createGrid("medium"), mediumMatrixSize.length);
})

it ('Correctly creates the grid for hard mode', () => {
    assert.equal(operations.createGrid("hard"), hardMatrixSize.length);
})
```

Figure 1: Unit test in test.js for user story 101



```
easyMatrixSize = [
    [1,1,1],
    [1,1,1],
    [1,1,1],
];

mediumMatrixSize = [
    [1,1,1,1,1,1],
    [1,1,1,1,1,1],
    [1,1,1,1,1,1],
    [1,1,1,1,1,1],
    [1,1,1,1,1,1],
    [1,1,1,1,1,1],
];

hardMatrixSize = [
    [1,1,1,1,1,1,1,1,1],
    [1,1,1,1,1,1,1,1,1],
    [1,1,1,1,1,1,1,1,1],
    [1,1,1,1,1,1,1,1,1],
    [1,1,1,1,1,1,1,1,1],
    [1,1,1,1,1,1,1,1,1],
    [1,1,1,1,1,1,1,1,1],
    [1,1,1,1,1,1,1,1,1],
    [1,1,1,1,1,1,1,1,1],
];
```

Figure 2: Matrices in test.js used to see if the output of the unit test matches with its respective grid length

```
var size;
var matrix = [];
length = 0;
if (x === "easy"){
    size = 3;

    for(var i = 0; i < size; i++){
        matrix[i] = [];
        for (var j = 0; j < size; j++){
            var random = Math.floor(Math.random() * 10);
            var num;
            switch(random){
                case 0:
                    num = '';
                    break;
                case 1:
                    num = '1';
                    break;
                case 2:
                    num = '2';
                    break;
                case 3:
                    num = '3';
                    break;
                case 4:
                    num = '4';
                    break;
                case 5:
                    num = '5';
                    break;
                case 6:
                    num = '6';
                    break;
                case 7:
                    num = '7';
                    break;
                case 8:
                    num = '8';
                    break;
                case 9:
                    num = '9';
                    break;
            }
                matrix[i][j] = num;
        }
    }

    return matrix.length
```

Figure 3: Code used in userStory101.js to fill in the matrix with random numbers

Explanation

The the unit test will pass in a string of either “easy”, “medium” or “hard” which is used to represent what game mode was selected. This string will then get passed into a function located in userStory101.js called “createGrid” which takes in the parameter ‘x’. ‘x’ is then used to go into the correct if statement which will then set another variable of size equal to its corresponding difficulty. Next, two for loops will fill a 2D array with random numbers after which the length of that array will be returned back to test.js which will then be compared against the correct 2D array size.

Result