

CPSC 349: Front-End Web Engineering – Spring 2020

Homework 8, due 6 May 2020

In the Tic-Tac-Toe tutorial you followed at the Reactjs.org site, you built a Tic-Tac-Toe game that calculated who was the winner, disallowed moves after someone had won, and had a time machine that allowed a user to back up and try different combinations of moves.

In doing so, you learned some basic features about ReactJS. The game, though, lacked some other basic features, such as:

1. It only allowed 3 x 3 grids (which IS the standard).
2. it didn't list the coordinates of each move (row, col).
3. It didn't **bold** the currently selected move in the move list.
4. It hardcoded the creation of the grid, instead of using two for loops.
5. It sorted the moves in descending order (which actually makes sense most of the time) It didn't highlight the squares that caused the winning move.
6. It didn't display a message about draws (aka Cats' games).

In this homework assignment, you will correct each of these deficiencies. Note, they are listed in order of increasing difficulty.

For extra credit, you can add AI to your game so the computer plays against you. If you allow the computer to go first, the game will be more difficult for you to win.

Submission

Turn in the code for this homework by uploading your code to a public Github repository. You may discuss this homework assignment with others, however, the work you submit must have been completed by own.

To complete your submission, **print the following sheet**, fill out the spaces below, and submit it to the professor in class by the deadline. Failure to follow the instructions exactly will incur a 10% penalty on the grade for this assignment.

CPSC 349 Homework Submission 8, due 6 May 2020

Your name:

Portfolio: https://theerrormaster.github.io

Verify each of the following items and place a checkmark in the correct column. Each item incorrectly marked will incur a 5% penalty on the grade for this assignment

Finished	Not finished	
yes		Prompt the user for the size of the Tic-Tac-Toe game and use that size
yes		List the coordinates of each move in your Time machine in (row, col) format.
yes		Bold the currently selected move in the Time machine.
yes		Use two for loops to create the grid, instead of hardcoding it.
yes		Allow the option of sorting the Time machine in ascending/descending order.
yes		Highlight the squares that caused the winning move in red.
yes		Display the appropriate message if the game ends in a draw.
	no	Add AI to your game so the computer plays against you. (extra credit) .
yes		Your game runs with no run-time errors in the React developer tools console
webpack		Your code was built using Parcel.
yes		Push your code to a public Github repository.

Comments:

Fill out and print this page, and submit it on the day this project is due.