5.0 hours

Project 4

Tic-Tac-Toe

Why can’t I submit this project yet?

You must first complete all prior Techdegree content before you can submit this project. Additionally, you cannot submit more than one project at a time.

Submit for Peer Review

Your current activity is HTML Tables.

- Instructions
- How you'll be graded

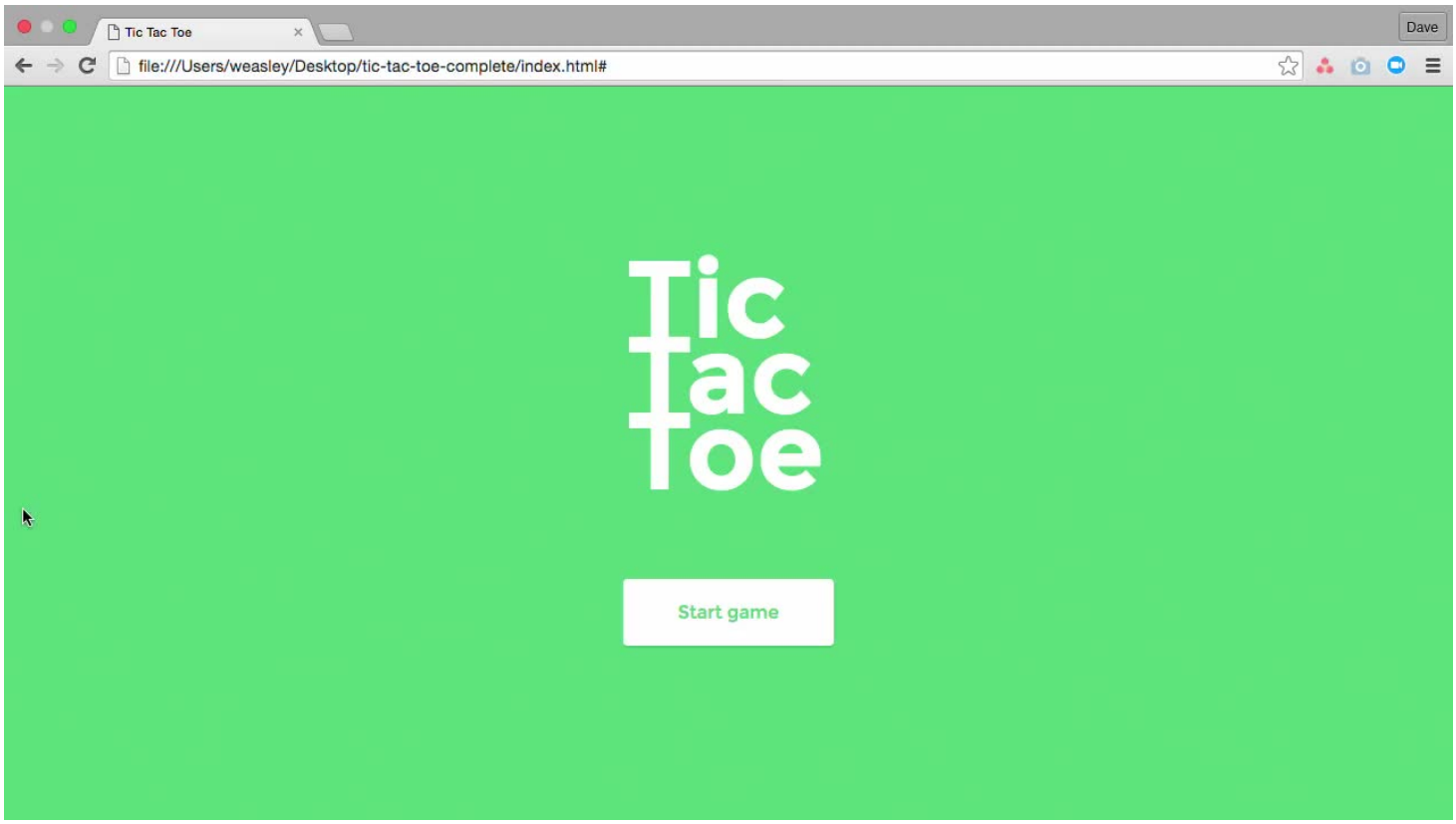
In this project, you'll build a functional, two-person Tic Tac Toe game. You'll use the provided mockups, HTML, CSS and image files to create a game that requires players to add their names, take turns adding an X or O to the game board, and announce when the game ends. You'll need to keep track of the state of the game -- whose turn it is, where the X's and O's are on the board, and whether the game is a draw or, if not, who won and lost.

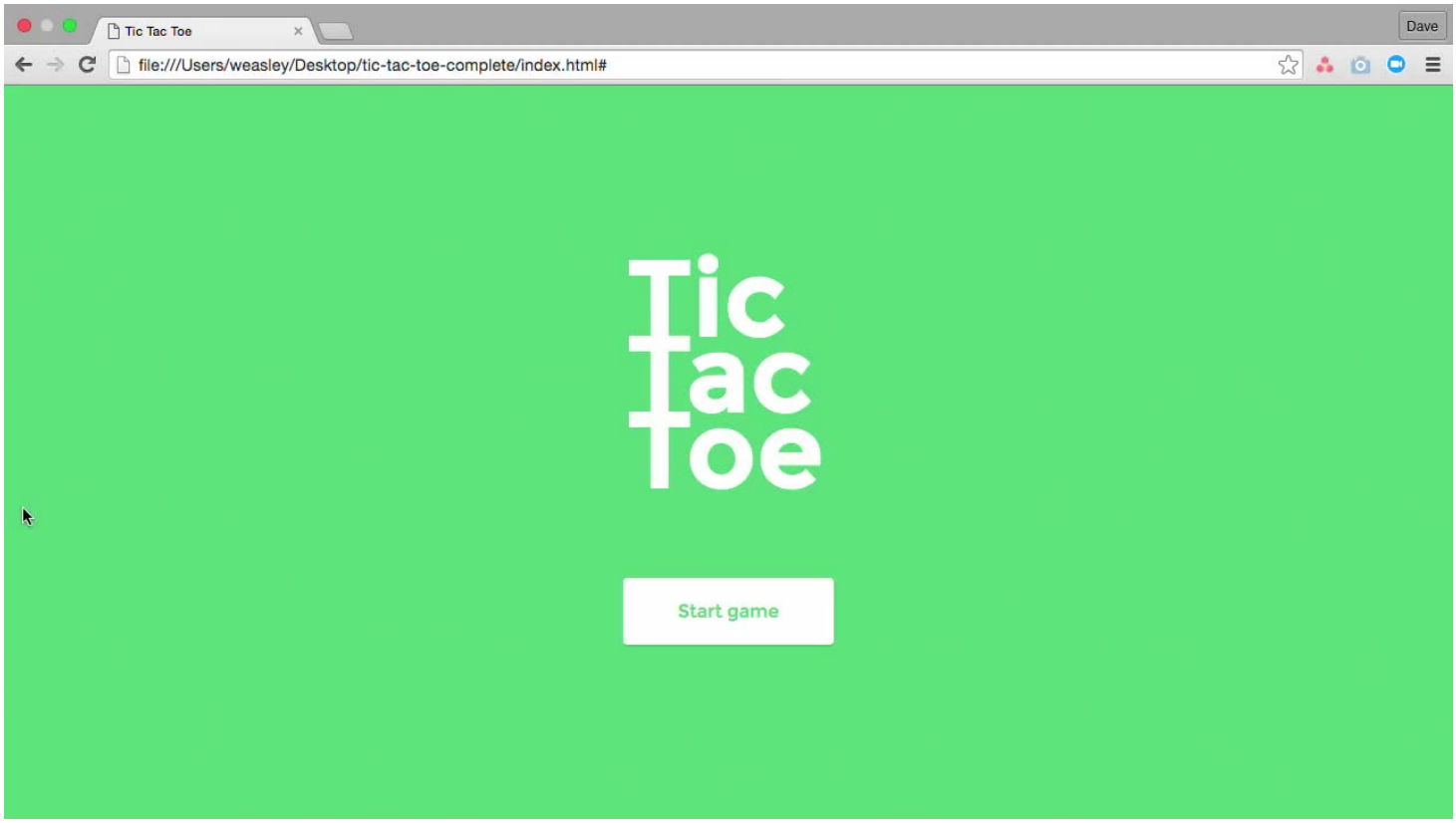
You'll use your knowledge of JavaScript data structures like arrays and objects as well as DOM-manipulation using jQuery or plain JavaScript to complete this project.

And, to ensure that you're using good programming practices, you’ll use the module pattern to create your Tic-Tac-Toe game. In other words, you should wrap all of your code in a single global variable, or execute it all in a single self-invoking function. See the link in the project resources for a Treehouse workshop on the module pattern.

NOTE: To get an "Exceeds Expectations" grade for this project, you'll need to "exceed" on **every** requirement that has an "Exceeds Expectations" option.

Video Player


Tic Tac Toe
Dave
file:///Users/weasley/Desktop/tic-tac-toe-complete/index.html#
Tic
Tac
Toe
Start game


Tic Tac Toe
Dave
file:///Users/weasley/Desktop/tic-tac-toe-complete/index.html#
Tic
Tac
Toe
Start game

Before you start

To prepare for this project you'll need to make sure you complete and understand these steps.

2 steps

- **Make sure you have a GitHub account and know how to create a new repository and upload files to it. If you need a reminder on how to use GitHub and GitHub desktop, check out the workshop Share Your Projects wIth GitHub Desktop.**

- **If you need a reminder on how to use GitHub and GitHub desktop to create a new repository check out the workshop 'Share Your Projects wIth GitHub Desktop' in the Project Resources.**

  Download the project files. We've provided several files to help with this project:

  - The `mockups` folder shows what the page should look like when it first loads, as the game is being played, and when the game is over.
  - The `html_snippets` folder contains three files: `start.txt` includes the HTML for the opening screen. What players see when the page first loads. `board.txt` includes the HTML of the Tic Tac Toe gameboard. `win.txt` includes the HTML to display when the game is over. You'll need to modify this HTML for when player 0 or X wins or when there's a tie
  - The `css` folder contains the CSS for the game
  - The `img` folder contains two SVG images -- `o.svg` and `x.svg`. These images are used to display the players, and are used in the boxes of the board to indicate who claimed that square
    - The `index.html` file is the page where the game will be displayed
    - The `js` folder is empty -- that's where you'll add your JavaScript file or files.

Project Instructions

To complete this project, follow the instructions below. If you get stuck, ask a question in the community.

7 steps

- **Use the supplied mockup files and HTML snippets to guide you in building a Tic Tac Toe game. You can use jQuery or plain JavaScript to complete this project. Don't use an already programmed Tic Tac Toe plugin or library.**

- **When the page loads, the startup screen should appear. Use the `tictactoe-01-start.png` mockup, and the `start.txt` HTML snippet to guide you.**

- **Add programming, so that when the player clicks the start button the start screen disappears, the board appears, and the game begins. Use the `tictactoe-02-inprogress.png` mockup, and the `board.txt` HTML snippet to guide you.**

- **Add the game play following these rules:**

    - Play alternates between X and O.
    - The current player is indicated at the top of the page -- the box with the symbol O or X is highlighted for the current player. You can do this by simply adding the class `.active` to the proper list item in the HTML. For example, if it's player one's turn, the HTML should look like this: `<li class="players active" id="player1">`
    - When the current player mouses over an empty square on the board, it's symbol the X or O should appear on the square. You can do this using the `x.svg` or `o.svg` graphics (hint use JavaScript to set the background-image property for that box.)
    - Players can only click on empty squares. When the player clicks on an empty square, attach the class `box-filled-1` (for O) or `box-filled-2` (for X) to the square. The CSS we're providing will automatically add the proper image to the square marking it as occupied.
    - The game ends when one player has three of their symbols in a row either horizontally, vertically or diagonally. If all of the squares are filled and no players have three in a row, the game is a tie.

- **Add programming so that when the game ends, the board disappears and the game end screen appears. Use the `tictactoe-03-winner1.png` and `tictactoe-04-winner2.png` mockups, and the `win.txt` HTML snippet for guidance. Depending on the game results the final screen should:**

    - Show the word "Winner" or the phrase "It's a Tie!"
    - Add the appropriate class to the `<div>` for the winning screen: `<div class="screen screen-win" id="finish">` `screen-win-one` for player 1, `screen-win-two` for player two, or `screen-win-tie` if the game ends with no winner. For example, if player 1 wins, the HTML should look like this: `<div class="screen screen-win screen-win-one" id="finish">`

- **Add programming so that when a player pushes the "New Game" button, the board appears again, empty, and a new game begins.**

- **Use the module pattern to wrap all of your JavaScript code into a single global variable or an immediately invoked function.**

Extra Credit

To get an "exceeds" rating, you can expand on the project in the following ways:

4 steps

- **On the start screen, prompt the user add their name before the game starts**
- **Display the player's name on the board screen during game play**
- **Add programming to support playing against the computer. Only one player plays; the other is controlled by your programming.**
- **Display the player's name if they win the game**

NOTE:

- To get an "Exceeds Expectations" grade for this project, you'll need to complete **each** of the items in this section. See the rubric in the "**How You'll Be Graded**" tab above for details on how you'll be graded.
- If you're shooting for the "Exceeds Expectations" grade, it is recommended that you mention so in your submission notes.
- Passing grades are final. If you try for the "Exceeds Expectations" grade, but miss an item and receive a "Meets Expectations" grade, you won't get a second chance. Exceptions can be made for items that have been misgraded in review.

Download files

Zip file

Project Resources

External Link

Tic-tac-toe

External Link

Understanding the Minimax Algorithm

Workshop

The Module Pattern in JavaScript

External Link

Immediately Invoked Function Expressions

External Link

Bookmarklet to Detect Global Variables

Need Help?

Have questions about this project? Start a discussion with the community and Treehouse staff.

Get Help