



How to build a Tic-Tac-Toe Game using React Hooks ?



tejsidda34

Read

Discuss

Courses

Practice

Video

React is a frontend open-source JavaScript library, used to build interactive User Interfaces. React is focused on Single Page applications and is more popularly known as a SPA. In this tutorial, we'll use React and its hooks to build a fun Tic-Tac-Toe application. Before jumping into code make sure the pre-requisites are checked for a better understanding.

Prerequisite:

- [Introduction to React](#)
- [Functional Components in React](#)
- [React Hooks](#)

Modules required:

- npm
- React

Creating React App and Setting Up:

Step 1: You will start a new project using [create-react-app](#) so open your terminal and type.

```
npx create-react-app tic-tac-toe-react
```

Step 2: Switch to the tic-tac-toe-react folder using the following command.

```
cd tic-tac-toe-react
```

Step 3: Change to the src folder and remove the unnecessary stuff using

the following command

```
cd src
rm *
```

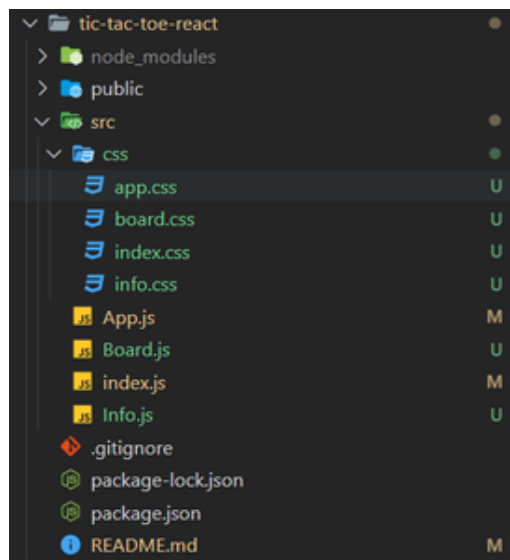
Step 4: Create a **css folder** in src, which contains the app.css, board.css, index.css, and info.css files.

```
mkdir css
touch app.css board.css index.css info.css
```

Step 5: In the **src folder**, create App.js, Board.js, index.js, and Info.js files.

```
touch App.js Board.js index.js Info.js
```

Project Structure: The file structure in the project will look like this.



Example: This example will guide you with code to build a Tic-Tac-Toe game using React Hooks.

index.js: This file links the HTML file and the react code. Edit the index.js file in the following manner:

index.js

```
import React from 'react';
import ReactDOM from 'react-dom';
import './css/index.css';
import App from './App';
```

```
ReactDOM.render(
```

```
    <React.StrictMode>
      <App />
    </React.StrictMode>,
    document.getElementById('root')
  );
```

App.js: This file acts like a base file containing the Info and Board components. Edit the App.js file in the following manner:

App.js

```
// Importing the required components
import Board from './Board';
import Info from './Info';

// Importing the CSS File
import './css/app.css';

// Importing the useState hook
import { useState } from 'react';

function App() {

  // Creating a reset state, which indicates whether
  // the game should be reset or not
  const [reset, setReset] = useState(false);

  // Creating a winner state, which indicates
  // the current winner
  const [winner, setWinner] = useState('');

  // Sets the reset property to true
  // which starts the chain
  // reaction of resetting the board
  const resetBoard = () => {
    setReset(true);
  }

  return (
    <div className="App">
      {/* Shrinks the popup when there is no winner */}
      <div className={`winner ${winner !== '' ? '' : 'shrink'}`} `
        {/* Display the current winner */}
        <div className="winner-text">{winner}</div>
        {/* Button used to reset the board */}
        <button onClick={() => resetBoard()}>
          Reset Board
        </button>
      </div>
```

```

        {/* Custom made board component comprising of
        the tic-tac-toe board */}
        <Board reset={reset} setReset={setReset} winner={winner}
            setWinner={setWinner} />
        <Info />
    </div>
    );
}

export default App;

```

Board.js: This file contains the tic-tac-toe board and the game logic. Edit the Board.js in the following manner:

Board.js

```

// Importing the CSS for the board
import './css/board.css';

// Importing the useState hook, useEffect hook and useRef hook
import { useState, useEffect, useRef } from 'react';

const Board = ({ reset, setReset, winner, setWinner }) => {

    // Creating a turn state, which indicates the current turn
    const [turn, setTurn] = useState(0);

    // Creating a data state, which contains the
    // current picture of the board
    const [data, setData] = useState(['', '', '', '', '',
        '', '', '', '']);

    // Creating a reference for the board
    const boardRef = useRef(null);

    // Function to draw on the board
    const draw = (event, index) => {
        // Draws only if the position is not taken
        // and winner is not decided yet
        if (data[index - 1] === '' && winner === '') {

            // Draws X if it's player 1's turn else draws 0
            const current = turn === 0 ? 'X' : '0'

            // Updating the data state
            data[index - 1] = current;

            //Drawing on the board
            event.target.innerText = current;

```

```

        // Switching the turn
        setTurn(turn === 0 ? 1 : 0)
    }
}

// useEffect hook used to reset the board whenever
// a winner is decided
useEffect(() => {

    // Clearing the data state
    setData(['', '', '', '', '', '', '', '', '']);

    // Getting all the children(cells) of the board
    const cells = boardRef.current.children

    // Clearing out the board
    for (let i = 0; i < 9; i++) {
        cells[i].innerText = '';
    }

    // Resetting the turn to player 0
    setTurn(0);

    // Resetting the winner
    setWinner('');
    setReset(false);
}, [reset, setReset, setWinner])

// useEffect hook used to check for a winner
useEffect(() => {

    // Checks for the win condition in rows
    const checkRow = () => {
        let ans = false;
        for (let i = 0; i < 9; i += 3) {
            ans |= (data[i] === data[i + 1] &&
                data[i] === data[i + 2] &&
                data[i] !== '')
        }
        return ans;
    }

    // Checks for the win condition in cols
    const checkCol = () => {
        let ans = false;
        for (let i = 0; i < 3; i++) {
            ans |= (data[i] === data[i + 3] &&
                data[i] === data[i + 6] &&
                data[i] !== '')
        }
    }

```

```
        return ans;
    }

    // Checks for the win condition in diagonals
    const checkDiagonal = () => {
        return ((data[0] === data[4] &&
            data[0] === data[8] && data[0] !== '') ||
            (data[2] === data[4] && data[2] === data[6] &&
            data[2] !== ''));
    }

    // Checks if at all a win condition is present
    const checkWin = () => {
        return (checkRow() || checkCol() || checkDiagonal());
    }

    // Checks for a tie
    const checkTie = () => {
        let count = 0;
        data.forEach((cell) => {
            if (cell !== '') {
                count++;
            }
        })
        return count === 9;
    }

    // Setting the winner in case of a win
    if (checkWin()) {
        setWinner(turn === 0 ? "Player 2 Wins!" :
            "Player 1 Wins!");
    } else if (checkTie()) {

        // Setting the winner to tie in case of a tie
        setWinner("It's a Tie!");
    }

    })

    return (
        <div ref={boardRef} className="board">
            <div className="input input-1"
                onClick={e => draw(e, 1)}></div>
            <div className="input input-2"
                onClick={e => draw(e, 2)}></div>
            <div className="input input-3"
                onClick={e => draw(e, 3)}></div>
            <div className="input input-4"
                onClick={e => draw(e, 4)}></div>
            <div className="input input-5"
                onClick={e => draw(e, 5)}></div>
            <div className="input input-6"
```

```

        onClick={e => draw(e, 6)}></div>
      <div className="input input-7"
        onClick={e => draw(e, 7)}></div>
      <div className="input input-8"
        onClick={e => draw(e, 8)}></div>
      <div className="input input-9"
        onClick={e => draw(e, 9)}></div>
    </div>
  )
}

```

Info.js: This file contains info about the tic-tac-toe game. Edit Info.js in the following manner:

Info.js

```

// Importing the css for the info
import "../css/info.css";

const Info = () => {
  return (
    <div className="info">
      <div className="player">Player 1: X</div>
      <div className="player">Player 2: O</div>
    </div>
  )
}

export default Info;

```

index.css

```

*{
  -webkit-box-sizing: border-box;
  -moz-box-sizing: border-box;
  box-sizing: border-box;
}

body {
  margin: 0;
  font-family: -apple-system, BlinkMacSystemFont, 'Segoe UI',
    'Roboto', 'Oxygen', 'Ubuntu', 'Cantarell', 'Fira Sans',
    'Droid Sans', 'Helvetica Neue',
    sans-serif;
  -webkit-font-smoothing: antialiased;
  -moz-osx-font-smoothing: grayscale;
}

```

```
}  
  
code {  
  font-family: source-code-pro, Menlo, Monaco,  
    Consolas, 'Courier New',  
    monospace;  
}
```

App.css

```
@import url(  
  'https://fonts.googleapis.com/css2?family=Bellefair&display=swap');  
  
.App{  
  width: 100vw;  
  height: 100vh;  
  display: flex;  
  justify-content: center;  
  align-items: center;  
  flex-direction: column;  
  gap: 5vh;  
  backdrop-filter: 5px;  
  background-color: #101010;  
}  
  
.winner {  
  transition: all ease-in .3s;  
  display: flex;  
  opacity: 1;  
  font-size: 1.5rem;  
  font-weight: 600;  
  gap: 1vh;  
  flex-direction: column;  
  justify-content: center;  
  align-items: center;  
  width: 20vw;  
  position: absolute;  
  top: 50%;  
  left: 50%;  
  transform: translate(-50%, -70%);  
  background-color: rgba(195, 141, 158, 0.863);  
  backdrop-filter: 5px;  
  padding: .5rem;  
  padding-bottom: 1rem;  
  border-radius: 10%;  
}  
  
.winner-text{  
  padding: .3em 1em .25em;
```



```
    font-weight: 600;
    font-size: 2.5rem;
    color: white;
    font-family: 'Bellefair', serif;
    position: relative;
    text-align: center;
    line-height: 1.3;
}

.shrink {
  transform: scale(.1);
  opacity: 0;
}

button {
  background-color: #111827;
  border: 1px solid transparent;
  border-radius: .75rem;
  box-sizing: border-box;
  color: #FFFFFF;
  cursor: pointer;
  flex: 0 0 auto;
  font-family: "Inter var";
  font-size: 1.125rem;
  font-weight: 600;
  line-height: 1.5rem;
  padding: .75rem 1.2rem;
  text-align: center;
  text-decoration: none #6B7280 solid;
  text-decoration-thickness: auto;
  transition-duration: .2s;
  transition-property: background-color, border-color,
    color, fill, stroke;
  transition-timing-function: cubic-bezier(.4, 0, 0.2, 1);
  user-select: none;
  -webkit-user-select: none;
  touch-action: manipulation;
  width: auto;
}

button:hover {
  background-color: #374151;
}

button:focus {
  box-shadow: none;
  outline: 2px solid transparent;
  outline-offset: 2px;
}

@media (min-width: 768px) {
  button {
    padding: .75rem 1.5rem;
  }
}
```

```
    }  
  };  
};
```

board.css

```
:root {  
  --board-background: none;  
  --border-color: #f6546a;  
  --border-thickness: 5px;  
}  
  
.board{  
  width: 30vw;  
  height: 50%;  
  background-color: var(--board-background);  
  display: flex;  
  align-items: flex-start;  
  flex-direction: row;  
  flex-wrap: wrap;  
}  
  
.input{  
  height: 33.33%;  
  width: 33.33%;  
  display: flex;  
  justify-content: center;  
  align-items: center;  
  color: whitesmoke;  
  font-family: 'Bellefair', serif;  
  font-style: italic;  
  font-weight: 700;  
  font-size: 6rem;  
}  
  
.input-1{  
  border-right: var(--border-thickness)  
    dashed var(--border-color);  
  border-bottom: var(--border-thickness)  
    dashed var(--border-color);  
}  
  
.input-2{  
  border-right: var(--border-thickness)  
    dashed var(--border-color);  
  border-bottom: var(--border-thickness)  
    dashed var(--border-color);  
}  
  
.input-3{  
  border-bottom: var(--border-thickness)
```

```
        dashed var(--border-color);
    }

    .input-4{
        border-right: var(--border-thickness)
            dashed var(--border-color);
        border-bottom: var(--border-thickness)
            dashed var(--border-color);
    }

    .input-5{
        border-right: var(--border-thickness)
            dashed var(--border-color);
        border-bottom: var(--border-thickness)
            dashed var(--border-color);
    }

    .input-6{
        border-bottom: var(--border-thickness)
            dashed var(--border-color);
    }

    .input-7{
        border-right: var(--border-thickness)
            dashed var(--border-color);
    }

    .input-8{
        border-right: var(--border-thickness)
            dashed var(--border-color);
    }
}
```

info.css

```
.info {
    width: 30vw;
    display: flex;
    justify-content: space-evenly;
    align-items: center;
    color: whitesmoke;
}

.player {
    border: 2px solid #f6546a;
    border-radius: 5%;
    padding: .5rem 0;
    display: flex;
    font-size: 1.5rem;
    justify-content: center;
    align-items: center;
}
```

```
    width: 10vw;  
  }
```

Save all files and start the application by running the following command:

```
npm start
```

Output:



Last Updated : 20 Apr, 2022

6

Similar Reads

1. Explain the new feature of React hooks introduced in React v16.8

2. How to generate random colors by using React hooks ?

3. React-Router Hooks

4. React Suite Notification Props & Hooks

5. Things You Should Know About React Hooks

6. How to use componentWillMount() in React Hooks?

7. Introduction to React Hooks

8. ReactJS Hooks Complete Reference

9. What are hooks and when we use them ?

10. How to define JavaScript Hooks ?

Related Tutorials

1. Onsen UI

2. React Material UI

3. NuxtJS

4. D3.js

5. Spectre CSS

Previous

Semantic-UI Item Types

Next

**Explain the arrow function syntax in
TypeScript**

Article Contributed By :



tejsidda34

tejsidda34

Vote for difficulty

Current difficulty : [Easy](#)

Easy

Normal

Medium

Hard

Expert

Improved By : [nikhatkhan11](#)

Article Tags : [CSS-Properties](#), [Picked](#), [React-Hooks](#), [React-Questions](#), [ReactJS](#),
[Web Technologies](#)

Improve Article

Report Issue



A-143, 9th Floor, Sovereign Corporate
Tower, Sector-136, Noida, Uttar Pradesh -
201305

feedback@geeksforgeeks.org

Company

[About Us](#)

[Careers](#)

[In Media](#)

[Contact Us](#)

[Terms and Conditions](#)

[Privacy Policy](#)

[Copyright Policy](#)

[Third-Party Copyright Notices](#)

[Advertise with us](#)

Explore

[Job Fair For Students](#)

[POTD: Revamped](#)

[Python Backend LIVE](#)

[Android App Development](#)

[DevOps LIVE](#)

[DSA in JavaScript](#)

Languages

[Python](#)

[Java](#)

[C++](#)

[GoLang](#)

[SQL](#)

[R Language](#)

[Android Tutorial](#)

Data Structures

[Array](#)

[String](#)

[Linked List](#)

[Stack](#)

[Queue](#)

[Tree](#)

[Graph](#)

Algorithms

[Sorting](#)

[Searching](#)

[Greedy](#)

[Dynamic Programming](#)

[Pattern Searching](#)

[Recursion](#)

[Backtracking](#)

Web Development

[HTML](#)

[CSS](#)

[JavaScript](#)

[Bootstrap](#)

[ReactJS](#)

[AngularJS](#)

[NodeJS](#)

Computer Science

GATE CS Notes
Operating Systems
Computer Network
Database Management System
Software Engineering
Digital Logic Design
Engineering Maths

Data Science & ML

Data Science With Python
Data Science For Beginner
Machine Learning Tutorial
Maths For Machine Learning
Pandas Tutorial
NumPy Tutorial
NLP Tutorial
Deep Learning Tutorial

Competitive Programming

Top DSA for CP
Top 50 Tree Problems
Top 50 Graph Problems
Top 50 Array Problems
Top 50 String Problems
Top 50 DP Problems
Top 15 Websites for CP

Interview Corner

Company Preparation
Preparation for SDE
Company Interview Corner
Experienced Interview
Internship Interview
Competitive Programming
Aptitude

Python

Python Programming Examples
Django Tutorial
Python Projects
Python Tkinter
OpenCV Python Tutorial
Python Interview Question

DevOps

Git
AWS
Docker
Kubernetes
Azure
GCP

System Design

What is System Design
Monolithic and Distributed SD
Scalability in SD
Databases in SD
High Level Design or HLD
Low Level Design or LLD
Top SD Interview Questions

GfG School

CBSE Notes for Class 8
CBSE Notes for Class 9
CBSE Notes for Class 10
CBSE Notes for Class 11
CBSE Notes for Class 12
English Grammar

Commerce

Accountancy
Business Studies
Microeconomics
Macroeconomics
Statistics for Economics
Indian Economic Development

SSC/ BANKING

SSC CGL Syllabus
SBI PO Syllabus
SBI Clerk Syllabus
IBPS PO Syllabus
IBPS Clerk Syllabus
Aptitude Questions
SSC CGL Practice Papers

UPSC

Polity Notes
Geography Notes
History Notes
Science and Technology Notes
Economics Notes
Important Topics in Ethics
UPSC Previous Year Papers

Write & Earn

Write an Article
Improve an Article
Pick Topics to Write
Write Interview Experience
Internships
Video Internship

@geeksforgeeks , Some rights reserved