

Міністерство освіти і науки України
Львівський національний університет імені Івана Франка
Факультет електроніки і комп'ютерних технологій

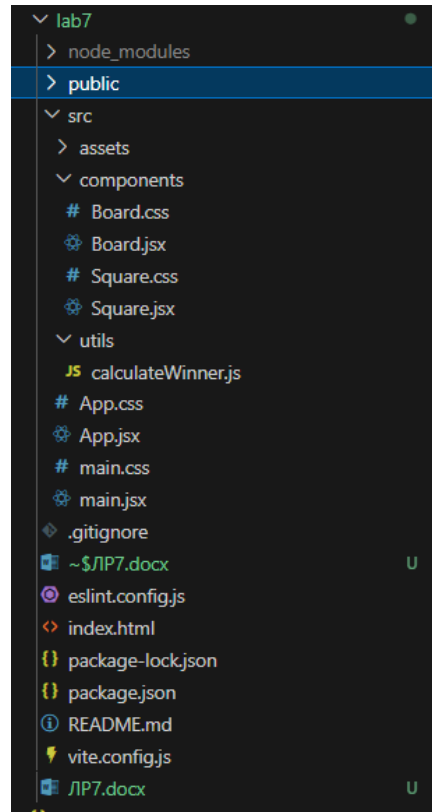
Лабораторна робота №6
З курсу “Веб програмування на стороні клієнта”
“Порівняння розробки ToDo List за допомогою чистого JavaScript та React”

Виконав:
студент ФЕІ-25
Кайда Матвій
Перевірив:
доц. Анохін В. М.

Львів 2025

Мета: Ознайомитися з основами розробки веб-додатків на React. Вивчити концепції компонентного підходу, пропсів (props) та стану (state). Реалізувати управління подіями, оновлення інтерфейсу та перевірку логіки гри.

Структура проекту:



Загальний вигляд:

Хрестики-Нулики

■ X: 4 | ■ O: 0 | ○ Нічий: 2

Хід гравця: X

Почати знову

Статистика зберігається в localStorage.

App.jsx:

```
import React from "react";
import "../App.css";
import Board from "../components/Board";

const App = () => {
  return (
    <div className="app-container">
      <Board />
    </div>
  );
};

export default App;
```

Board.jsx:

```
import React, { useEffect, useState } from "react";
import calculateWinner from "../utils/calculateWinner";
import "../Board.css";
import Square from "../Square";

const createEmptyBoard = () => Array(3).fill(null).map(() => Array(3).fill(null));

const Board = () => {
  const [board, setBoard] = useState(createEmptyBoard);
  const [xIsNext, setXIsNext] = useState(true);
  const [stats, setStats] = useState(() => {
    const saved = localStorage.getItem("ttt-stats");
    return saved ? JSON.parse(saved) : { X: 0, O: 0, draw: 0 };
  });

  const [winningCombo, setWinningCombo] = useState([]);

  useEffect(() => {
    const saved = localStorage.getItem("ttt-stats");
    if (saved) setStats(JSON.parse(saved));
  }, []);

  useEffect(() => {
    localStorage.setItem("ttt-stats", JSON.stringify(stats));
  }, [stats]);

  const winner = winningCombo.length ? board[winningCombo[0]][0][winningCombo[0][1]] : null;
  const isDraw = !winner && board.flat().every(Boolean);

  const status = winner
    ? `Переможець: ${winner}`
    : isDraw
    ? "Нічия!"
    : `Хід гравця: ${xIsNext ? "X" : "O"} `;

  const handleClick = (row, col) => {
    if (board[row][col] || winner) return;

    const newBoard = board.map((r) => [...r]);
    const current = xIsNext ? "X" : "O";
    newBoard[row][col] = current;


    setBoard(newBoard);
    setXIsNext(!xIsNext);

    const result = calculateWinner(newBoard, row, col);
    if (result) {
      setWinningCombo(result.combo);
      setStats((prev) => ({ ...prev, [current]: prev[current] + 1 }));
    } else if (newBoard.flat().every(Boolean)) {
      setStats((prev) => ({ ...prev, draw: prev.draw + 1 }));
    }
  };
};
```

```

const restart = () => {
  setBoard(createEmptyBoard());
  setXIsNext(true);
  setWinningCombo([]);
};

return (
  <div className="board-container">
    <h1 className="board-title">Хрестики-Нулики</h1>

    <div className="board-stats">
      |  X: {stats.X} |  O: {stats.O} |  Нічий: {stats.draw}
    </div>

    <table className="board-table">
      <tbody>
        {board.map((row, rowIndex) => (
          <tr key={rowIndex}>
            {row.map((value, colIndex) => (
              <td key={colIndex} className="board-cell">
                <Square value={value} onClick={() => handleClick(rowIndex, colIndex)} />
              </td>
            ))}
          </tr>
        ))}
      </tbody>
    </table>

    <div className="status-text">{status}</div>

    <button className="restart-button" onClick={restart}>
      | Почати знову
    </button>
  </div>
);
};

export default Board;

```

Square.jsx:

```

import React from "react";
import "./Square.css";

function Square({ value, onClick }) {
  return (
    <button className="square" onClick={onClick}>
      {value || <span className="invisible">-</span>}
    </button>
  );
}

export default Square;

```

calculateWinner.js:

```

const calculateWinner = (board, row, col) => {
  const player = board[row][col];
  const size = board.length;
  let winCombo = [];

  // перевірка рядка
  if (board[row].every((cell) => cell === player)) {
    winCombo = board[row].map((_, i) => [row, i]);
    return { winner: player, combo: winCombo };
  }

  // перевірка стовпця
  if (board.every((r) => r[col] === player)) {
    winCombo = board.map((_, i) => [i, col]);
    return { winner: player, combo: winCombo };
  }

  // головна діагональ
  if (row === col && board.every((r, i) => r[i] === player)) {
    winCombo = board.map((_, i) => [i, i]);
    return { winner: player, combo: winCombo };
  }

  // побічна діагональ
  if (row + col === size - 1 && board.every((r, i) => r[size - 1 - i] === player)) {
    winCombo = board.map((_, i) => [i, size - 1 - i]);
    return { winner: player, combo: winCombo };
  }

  return null;
};

export default calculateWinner;

```

Висновок: Під час виконання було створено проект “Хрестики-нулики” відповідно вимогам. Properties було використано в компоненті Square, а стани компонент – в Board.jsx.