```jsx
import React, { useState } from "react";

import "../Game.css";

import Header from "./Header";
import Footer from "./Footer";
import GameCircle from "./GameCircle";


import { isDraw, isWinner, getComputerMove } from "../helper";
import {
    GAME_STATE_PLAYING,
    GAME_STATE_WIN,
    NO_PLAYER,
    PLAYER_1,
    PLAYER_2,
    NO_CIRCLES,
    GAME_STATE_DRAW,
} from "../Constants";

const GameBoard = () => {
    const [gameBoard, setGameBoard] = useState(Array(NO_CIRCLES).fill(NO_PLAYER));
    const [currentPlayer, setCurrentPlayer] = useState(PLAYER_1);
    const [gameState, setGameState] = useState(GAME_STATE_PLAYING);
    const [winPlayer, setWinPlayer] = useState(NO_PLAYER);


    const initGame = () => {
        setCurrentPlayer(PLAYER_1);
        setGameBoard(Array(NO_CIRCLES).fill(NO_PLAYER));
        setGameState(GAME_STATE_PLAYING);
    }

    const initBoard = () => {
        const circles = []; //array

        for (let i = 0; i < NO_CIRCLES; i++) {
            circles.push(renderCircle(i));
        }
        return circles;
    };

    const circleClicked = (id) => {
        console.log("circle clicked:" + id);

        if (gameBoard[id] !== NO_PLAYER) return;
        if (gameState !== GAME_STATE_PLAYING) return;

        if (isWinner(gameBoard, id, currentPlayer)) {
            setGameState(GAME_STATE_WIN);
            setWinPlayer(currentPlayer);
        }
        if (isDraw(gameBoard, id, currentPlayer)) {
            setGameState(GAME_STATE_DRAW);
            setWinPlayer(NO_PLAYER);
        }

        setGameBoard((prev) => {
            return prev.map((circle, pos) => {
                if (pos === id) return currentPlayer;
                return circle;
            });
        });

        setCurrentPlayer(currentPlayer === PLAYER_1 ? PLAYER_2 : PLAYER_1);

        console.log(gameBoard);
        console.log(currentPlayer);
```

Handwritten annotations (blue):
- ayyi function kay qdar yeqra valeurs d attributs li meah f html
- useState how getter w setter + init
- gameBoard(1)=0 etc
- hado yaeni default init vars
- 1 6 ... 0
- ki rej3o boyat w id player 0 teqdar tclicker f circle
- teqdar tclicker
- circles   with there id
- id   attribut f html
- may eml walo ida deja tclicka fih
- maykmlshi ida wqaf laeb
- f dik saea n testew wn emlo set direct ida true
- t afficha l winner f header, wiy wqaf laeb
- Array(gameboard) of circles(pos of circle)
- kayftesh f circles, f chaque pos, ida pos=id clicked, pos recoit current player
- return new updated gameBoard
- function kikon dial player1 matalan, takisali ad kan qelbo l player 2
- function li majji ghaykon d player 2

```
    };

    const renderCircle = (id) => {
        return (
            <GameCircle
                key={id} //daroria f react bash ki farraq
                id={id}
                className={`player_${gameBoard[id]}`}
                onCircleClicked={circleClicked}
            />
        );
    };

    const onSuggestClick = () => {
        circleClicked(getComputerMove(gameBoard));
    }

    const onNewGameClick = () => {
        initGame();
    }

    return (
        <>
            <Header gameState={gameState} currentPlayer={currentPlayer} winPlayer={winPlayer} />
            <div className="gameBoard">{initBoard()}</div>
            <Footer onSuggestClick={onSuggestClick} onNewGameClick={onNewGameClick} gameState=
{gameState}/>
        </>
    );
};

export default GameBoard;
```

Handwritten annotations:

1 2 3 4 emlna for foqi push

color   f had pos ghan affectiw id dial player

ki qdar yqra attributs kamlin 3adi

ghayclicker f circle li genera

PLAYING WIN DRAW bash kit affichaw   INIT 1 w 2

le plus important tjrs

circles liki tghayar b sebba d states dima ki tfaeal

init default values