JS:

/\* class names for win lines

<!-- Top  row line 1,2,3 = trl

        middle row line, 4,5,6 = mrl

        bottom row line, 7,8,9 = brl

        Left column line 1,4,7 = lcl

        middle column line, 2,5,8 = mcl

        right column line, 3,6,9 = rcl

        cross line from left top corner to

        right bottom corner, 1,5,9 =ltoprbottoml

        cross line from right top corner

        to left buttom corner, 3,5,7 = rtopbottomll--> \*/

/\* Boxes of the game \*/

const numb1 = document.querySelector(`.box1`);

const numb2 = document.querySelector(`.box2`);

const numb3 = document.querySelector(`.box3`);

const numb4 = document.querySelector(`.box4`);

const numb5 = document.querySelector(`.box5`);

const numb6 = document.querySelector(`.box6`);

const numb7 = document.querySelector(`.box7`);

const numb8 = document.querySelector(`.box8`);

const numb9 = document.querySelector(`.box9`);

//The game window

const tTT = document.querySelector(`.TTT`);

// game text

const winMsg = document.querySelector(`.winMsg`);

const startMsg = document.querySelector(`.startMsg`);

const replayMsg = document.querySelector(`.replayMsg`);

//Player representation

const player0 = `X`; // Red

const player1 = `O`; // Blue

//Switcher for player 1 and 2.

let playerTurn = true;

//boolean to avoid further selection of boxes

let gameOver = false;

/\* put the boxes intop array => Allows to

lopp trough the options. The plan was to use for loop like this:

Not possible because JS dosnt allow that.

let numbAll;

// Player 0 and 1, in red and blue

const selected0 = function () {

  for (let i = 0; i >= 9; i++)

    if (`numb${i}` === ``) {

      `numb${i}`.style.backgroundColor = `red`;

    }

};

const selected1 = function () {

  for (let i = 0; i >= 9; i++)

    if (`numb${i}` === ``) {

      `numb${i}`.style.backgroundColor = `blue`;

    }

}; \*/

let numbAll = [

  document.querySelector(`.box1`),

  document.querySelector(`.box2`),

  document.querySelector(`.box3`),

  document.querySelector(`.box4`),

  document.querySelector(`.box5`),

  document.querySelector(`.box6`),

  document.querySelector(`.box7`),

  document.querySelector(`.box8`),

  document.querySelector(`.box9`),

];

//Function to play the game text.

startMsg.classList.remove(`hidden`);

tTT.classList.add(`hidden`);

const playGameMsg = function () {

  startMsg.classList.add(`hidden`);

  tTT.classList.remove(`hidden`);

  replayMsg.classList.add(`hidden`);

  ticTacToeGame();

};

startMsg.addEventListener(`click`, playGameMsg);

//function to replay the game

const replayGame = function () {

  //This removes all the red and blue classes.

  numbAll.forEach((box) => {

    box.classList.remove("red");

    box.classList.remove("blue");

  });

  //hides the pop messages

  winMsg.classList.add(`hidden`);

  replayMsg.classList.add(`hidden`);

  //Resets the game state

  playerTurn = true;

  gameOver = false;

  //starts the game

  ticTacToeGame();

};

replayMsg.addEventListener(`click`, replayGame);

// funciton to stop the game and show the player who won.

winMsg.classList.add(`hidden`);

replayMsg.classList.add(`hidden`);

const gameWinnerMsg = function () {

  if (!playerTurn) {

    winMsg.textContent = `${player0} Won!`;

    winMsg.classList.remove(`hidden`);

    replayMsg.classList.remove(`hidden`);

    playerTurn = true;

  } else if (playerTurn) {

    winMsg.textContent = `${player1} Won!`;

    winMsg.classList.remove(`hidden`);

    replayMsg.classList.remove(`hidden`);

    playerTurn = false;

  }

  gameOver = true;

};

//The whole game in a callable function

const ticTacToeGame = function () {

  const changePlayer = function () {

    if (

      gameOver ||

      this.classList.contains("blue") ||

      this.classList.contains("red")

    ) {

      return;

    }

    if (playerTurn) {

      this.classList.add("blue");

      if (

        numb1.classList.contains("blue") &&

        numb2.classList.contains("blue") &&

        numb3.classList.contains("blue")

      ) {

        gameWinnerMsg();

      } else if (

        numb1.classList.contains("blue") &&

        numb5.classList.contains("blue") &&

        numb9.classList.contains("blue")

      ) {

        gameWinnerMsg();

      } else if (

        numb3.classList.contains("blue") &&

        numb5.classList.contains("blue") &&

        numb7.classList.contains("blue")

      ) {

        gameWinnerMsg();

      } else if (

        numb1.classList.contains("blue") &&

        numb4.classList.contains("blue") &&

        numb7.classList.contains("blue")

      ) {

        gameWinnerMsg();

      } else if (

        numb2.classList.contains("blue") &&

        numb5.classList.contains("blue") &&

        numb8.classList.contains("blue")

      ) {

        gameWinnerMsg();

      } else if (

        numb3.classList.contains("blue") &&

        numb6.classList.contains("blue") &&

        numb9.classList.contains("blue")

      ) {

        gameWinnerMsg();

      } else if (

        numb4.classList.contains("blue") &&

        numb5.classList.contains("blue") &&

        numb6.classList.contains("blue")

      ) {

        gameWinnerMsg();

      } else if (

        numb7.classList.contains("blue") &&

        numb8.classList.contains("blue") &&

        numb9.classList.contains("blue")

      ) {

        gameWinnerMsg();

      } else {

        console.log(`no1 won`);

      }

      playerTurn = false;

    } else {

      this.classList.add("red");

      if (

        numb1.classList.contains("red") &&

        numb2.classList.contains("red") &&

        numb3.classList.contains("red")

      ) {

        gameWinnerMsg();

      } else if (

        numb1.classList.contains("red") &&

        numb5.classList.contains("red") &&

        numb9.classList.contains("red")

      ) {

        gameWinnerMsg();

      } else if (

        numb3.classList.contains("red") &&

        numb5.classList.contains("red") &&

        numb7.classList.contains("red")

      ) {

        gameWinnerMsg();

      } else if (

        numb1.classList.contains("red") &&

        numb4.classList.contains("red") &&

        numb7.classList.contains("red")

      ) {

        gameWinnerMsg();

      } else if (

        numb2.classList.contains("red") &&

        numb5.classList.contains("red") &&

        numb8.classList.contains("red")

      ) {

        gameWinnerMsg();

      } else if (

        numb3.classList.contains("red") &&

        numb6.classList.contains("red") &&

        numb9.classList.contains("red")

      ) {

        gameWinnerMsg();

      } else if (

        numb4.classList.contains("red") &&

        numb5.classList.contains("red") &&

        numb6.classList.contains("red")

      ) {

        gameWinnerMsg();

      } else if (

        numb7.classList.contains("red") &&

        numb8.classList.contains("red") &&

        numb9.classList.contains("red")

      ) {

        gameWinnerMsg();

      }

      playerTurn = true;

    }

  };

  /\* Applies the switch \*/

  numbAll.forEach((box) => {

    box.addEventListener(`click`, changePlayer);

  });

};

CSS:

/\* class names for win lines

<!-- Top  row line 1,2,3 = trl

        middle row line, 4,5,6 = mrl

        bottom row line, 7,8,9 = brl

        Left column line 1,4,7 = lcl

        middle column line, 2,5,8 = mcl

        right column line, 3,6,9 = rcl

        cross line from left top corner to

        right bottom corner, 1,5,9 =ltoprbottoml

        cross line from right top corner

        to left buttom corner, 3,5,7 = rtopbottomll--> \*/

.TTT {

  background-color: aqua;

  width: 367px;

  height: 355px;

  position: absolute;

  top: 50%;

  left: 50%;

  transform: translate(-50%, -50%);

}

.boxBtn {

  font-size: 125.1111111px;

  margin: 5px 20px;

  border-radius: 5%;

  height: 90px;

  /\* Make the numbers invisible \*/

  color: transparent;

}

/\* Makes them side by side \*/

.row1,

.row2,

.row3 {

  display: flex;

}

.gameBox {

}

.gameStartMsg {

  display: flex;

  justify-content: center;

  align-items: center;

  height: 50vh;

}

.gameStartMsg button {

  background-color: transparent;

  border: none;

}

.startMsg {

  color: red;

  font-size: 50px;

  z-index: 999;

}

.startMsg:hover {

  scale: 1.5;

}

.winMsg {

  font-size: 150px;

  color: red;

}

.gameWonMsg {

  display: flex;

  justify-content: center;

  align-items: center;

}

.replayTxt {

  display: flex;

  justify-content: center;

  align-items: center;

  height: 1vh;

}

.hidden {

  display: none;

}

.boxBtn.blue {

  background-image: url("circle.png");

  background-repeat: no-repeat;

  background-size: contain; /\* adjust as needed \*/

}

.boxBtn.red {

  background-image: url("x.png");

  background-repeat: no-repeat;

  background-size: contain; /\* adjust as needed \*/

}

HTML:

<!DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="UTF-8" />

    <meta name="viewport" content="width=device-width, initial-scale=1.0" />

    <title>Tic Tac Toe</title>

    <link rel="stylesheet" href="./style.css" />

  </head>

  <body>

    <!-- TTT=Tic Tac Toe -->

    <section class="TTT">

      <!-- Top  row line 1,2,3 = trl

        middle row line, 4,5,6 = mrl

        bottom row line, 7,8,9 = brl

        Left column line 1,4,7 = lcl

        middle column line, 2,5,8 = mcl

        right column line, 3,6,9 = rcl

        cross line from left top corner to

        right bottom corner, 1,5,9 =ltoprbottoml

        cross line from right top corner

        to left buttom corner, 3,5,7 = rtopbottomll-->

      <div class="row1">

        <div class="gameBox">

          <button class="boxBtn box1 trl lcl ltoprbottoml">1</button>

        </div>

        <div class="gameBox">

          <button class="boxBtn box2 trl mcl">2</button>

        </div>

        <div class="gameBox">

          <button class="boxBtn box3 trl rcl rtopbottomll">3</button>

        </div>

      </div>

      <div class="row2">

        <div class="gameBox">

          <button class="boxBtn box4 mrl lcl">4</button>

        </div>

        <div class="gameBox">

          <button class="boxBtn box5 mrl mcl ltoprbottoml rtopbottomll">

            5

          </button>

        </div>

        <div class="gameBox">

          <button class="boxBtn box6 mrl rcl">6</button>

        </div>

      </div>

      <div class="row3">

        <div class="gameBox">

          <button class="boxBtn box7 brl lcl rtopbottomll">7</button>

        </div>

        <div class="gameBox">

          <button class="boxBtn box8 brl mcl">8</button>

        </div>

        <div class="gameBox">

          <button class="boxBtn box9 brl rcl ltoprbottoml">9</button>

        </div>

      </div>

    </section>

    <section class="gameTxt">

      <div class="gameWonMsg"><h1 class="winMsg">You won!</h1></div>

      <div class="gameStartMsg">

        <button>

          <h1 class="startMsg">Start Playing Tic Tac Toe!</h1>

        </button>

      </div>

      <!-- Same classes applied -->

      <div class="gameStartMsg replayTxt">

        <button><h1 class="startMsg replayMsg">Play Again!</h1></button>

      </div>

    </section>

    <script src="scirpt.js"></script>

  </body>

</html>