<!DOCTYPE html>

<html lang="pt-br">

<head>

<meta charset="UTF-8">

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

<style>

/\* Estilos do placar e dos jogadores \*/

body {

font-family: Arial, sans-serif;

text-align: center;

}

.player-card {

border: 1px solid #ddd;

padding: 10px;

margin: 10px;

display: inline-block;

}

.player-photo {

width: 100px;

height: 100px;

background-color: #f0f0f0;

margin-bottom: 10px;

}

.player-name {

font-size: 20px;

margin-bottom: 5px;

}

.buttons {

margin-top: 20px;

}

.button {

font-size: 18px;

padding: 10px 20px;

margin: 0 10px;

cursor: pointer;

transition: transform 0.5s;

}

.correct {

background-color: #00cc00;

}

.incorrect {

background-color: #ff0000;

}

.start-button {

background-color: #0000ff;

color: #ffffff;

}

.countdown {

font-size: 48px;

color: #ff0000;

margin-top: 20px;

}

/\* Animações para os botões \*/

@keyframes rocket {

0% { transform: translateY(0); }

100% { transform: translateY(-200px); }

}

@keyframes fire {

0% { background-color: #ff0000; }

100% { background-color: #ff4500; }

}

.correct:active {

animation: rocket 0.5s forwards;

}

.incorrect:active {

animation: fire 0.5s forwards;

}

</style>

</head>

<body>

<!-- Quadro do jogador -->

<div class="player-card" id="player1">

<div class="player-photo"></div>

<div class="player-name">Jogador 1</div>

<button class="button start-button">Start</button>

<button class="button correct">Certo</button>

<button class="button incorrect">Errado</button>

</div>

<!-- Repita o bloco acima para outros jogadores (Jogador 2 a Jogador 12) -->

<div class="countdown" id="countdown">10</div>

<script>

// Lógica para os botões e árvore genealógica

const players = document.querySelectorAll('.player-card');

let currentPlayerIndex = 0;

let timeLeft = 10;

const countdownElement = document.getElementById('countdown');

let countdownInterval;

players.forEach((player, index) => {

const startButton = player.querySelector('.start-button');

const correctButton = player.querySelector('.correct');

const incorrectButton = player.querySelector('.incorrect');

startButton.addEventListener('click', () => {

// Iniciar o jogo para o jogador atual

currentPlayerIndex = index;

startCountdown();

});

correctButton.addEventListener('click', () => {

// Avançar para o próximo nível

// Implementação personalizada para avançar o jogador

// e atualizar a árvore genealógica

resetCountdown();

});

incorrectButton.addEventListener('click', () => {

// Eliminar o jogador e passar para o próximo

player.style.display = 'none'; // Esconde o quadro do jogador

currentPlayerIndex = (currentPlayerIndex + 1) % players.length;

resetCountdown();

});

});

function startCountdown() {

// Iniciar a contagem regressiva

clearInterval(countdownInterval);

timeLeft = 10;

countdownElement.textContent = timeLeft;

countdownInterval = setInterval(() => {

timeLeft--;

countdownElement.textContent = timeLeft;

if (timeLeft < 0) {

clearInterval(countdownInterval);

countdownElement.textContent = 'Tempo Esgotado!';

// Implementação personalizada para finalizar o jogo para o jogador atual

}

}, 1000);

}

function resetCountdown() {

// Reiniciar a contagem regressiva para o próximo jogador

clearInterval(countdownInterval);

countdownElement.textContent = '10';

timeLeft = 10;

}

// Implementação da árvore genealógica

// Aqui você pode adicionar sua lógica para criar a árvore genealógica

// e gerenciar o fluxo do jogo até chegar ao vencedor.

</script>

</body>

</html>