

# Documentação do Jogo da Velha

**Aluno : Yago Gomes da Silva**

## Visão Geral

Este documento fornece uma visão geral e documentação para um simples jogo da velha implementado em JavaScript. O jogo suporta tanto o modo Jogador contra Jogador (PVP) quanto o modo Jogador contra Ambiente (PVE). A interface é construída usando HTML, CSS e JavaScript.

## Como Jogar

Abra o arquivo index.html em seu navegador .

Quando a página é carregada, um diálogo solicita ao usuário que escolha o modo de jogo (PVP ou PVE).

Após selecionar um modo, o tabuleiro do jogo é criado, e a área da mensagem de vitória é configurada.

O jogo começa, e os jogadores se revezam clicando nas células para colocar suas marcas (X ou Círculo).

O jogo termina quando um jogador vence ou o tabuleiro fica cheio (empate).

Os jogadores podem reiniciar o jogo clicando no botão "Reiniciar!".

## Estrutura do Projeto

O projeto consiste em várias funções e constantes:

### Constantes

**isCircleTurn:** Uma variável booleana para rastrear de quem é a vez (Círculo ou X).

**onPVE\_Mode:** Uma variável booleana indicando se o jogo está no modo Jogador contra Ambiente.

**winningCombinations:** Um array contendo arrays de combinações vencedoras no tabuleiro do jogo.

### Funções

**setQuerySelectors():** Recupera e retorna elementos DOM importantes usando seletores de consulta.

**playModeDialog():** Cria um diálogo para o usuário escolher o modo de jogo (PVP ou PVE).

**showPlayModeDialog():** Exibe o diálogo de seleção de modo e retorna uma Promise resolvida com o modo escolhido.

**makeBoard():** Cria dinamicamente o tabuleiro do jogo com 9 células.

**makeWinningMessage():** Cria um elemento div para exibir a mensagem de vitória.

**startGame():** Inicializa o estado do jogo, remove ouvintes de eventos e configura o tabuleiro inicial do jogo.

**endGame(isWin, isDraw):** Exibe a mensagem de vitória ou uma mensagem de empate com base no resultado do jogo.

**checkForWin(currentPlayer):** Verifica se o jogador atual venceu o jogo.

**checkForDraw():** Verifica se o jogo terminou em empate.

**setBoardHoverClass():** Define a classe de sobreposição no tabuleiro do jogo com base na vez do jogador atual.

**swapTurns():** Troca a vez do jogador atual.

**terminationConditions(classToAdd):** Verifica vitória ou empate e encerra o jogo conforme necessário.

**botPlay(cell):** Simula a jogada do bot no modo PVE.

**handleClick(event):** Manipula o evento de clique em uma célula, colocando a marca do jogador e acionando a jogada do bot no modo PVE, se aplicável. Inicialização

**playModeDialog():** Invocado para exibir o diálogo de seleção de modo quando o script é carregado.

**runGame():** Função assíncrona que inicializa o jogo com base no modo escolhido.

## Observações

O jogo utiliza ouvintes de eventos para responder aos cliques dos jogadores e atualiza dinamicamente o DOM com base no estado do jogo.

O bot no modo PVE faz jogadas aleatórias após o jogador fazer uma jogada.

O jogo pode ser reiniciado clicando no botão "Reiniciar!".

OBS: Só não foi possível incluir a funcionalidade do aumento das proporções do campo