

Linguagens Script

RELATÓRIO

“4 em Linha Especial”

- Diogo Franja Correia (2024152253)
- Miguel Gama da Silva Martins (2024138611)
- Tiago José Mendes Rodrigues Maninha (2024154842)

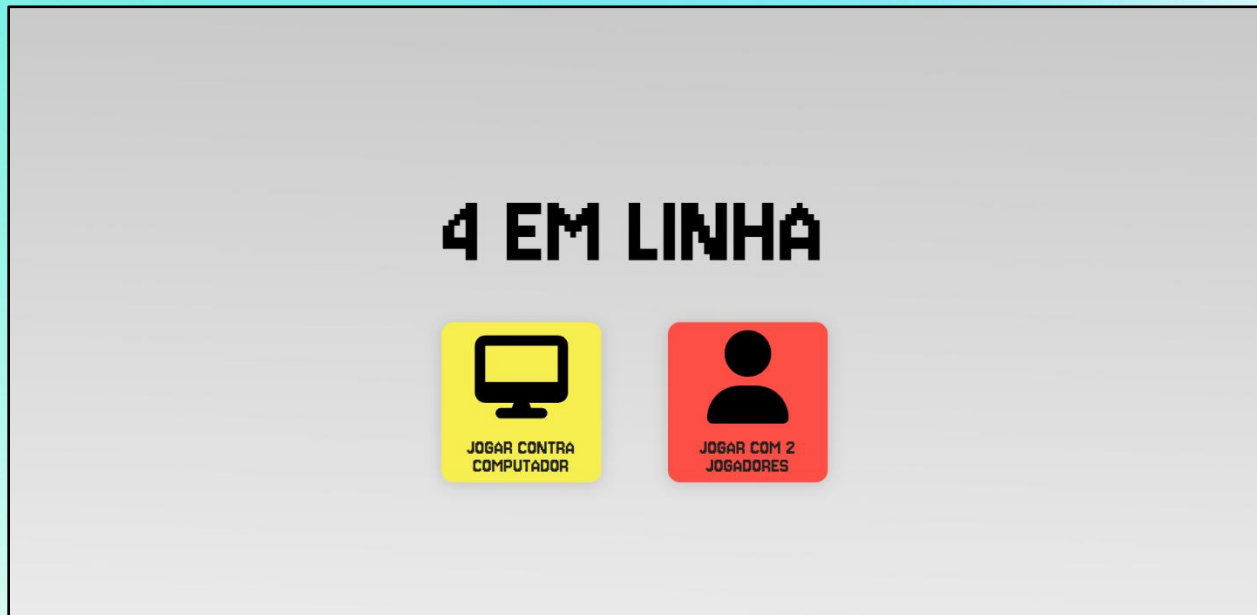


Figura 1 - Menu principal do jogo

RESUMO

O trabalho prático teve como objetivo o desenvolvimento de uma aplicação web em *ReactJS*, uma recriação personalizada do clássico jogo “4 em Linha”. Para a sua concretização, foram aplicados os conhecimentos adquiridos ao longo das aulas, com domínio das principais tecnologias envolvidas: *JavaScript*, *HTML* e *CSS*.

O jogo apresenta uma identidade visual especial, com uma temática alusiva às escolas do Instituto Politécnico de Coimbra (IPC), assim permitindo aos jogadores para além de escrever o nome de jogador, selecionar como sua equipa uma das seis escolas — ISEC, ISCAC, ESEC, ESAC, ESTeSC ou ESTGOH. Desta forma cria-se um visual que nos é familiar, promovendo também o espírito de interescolas, à semelhança do "IPC CUP".

1. Componentes

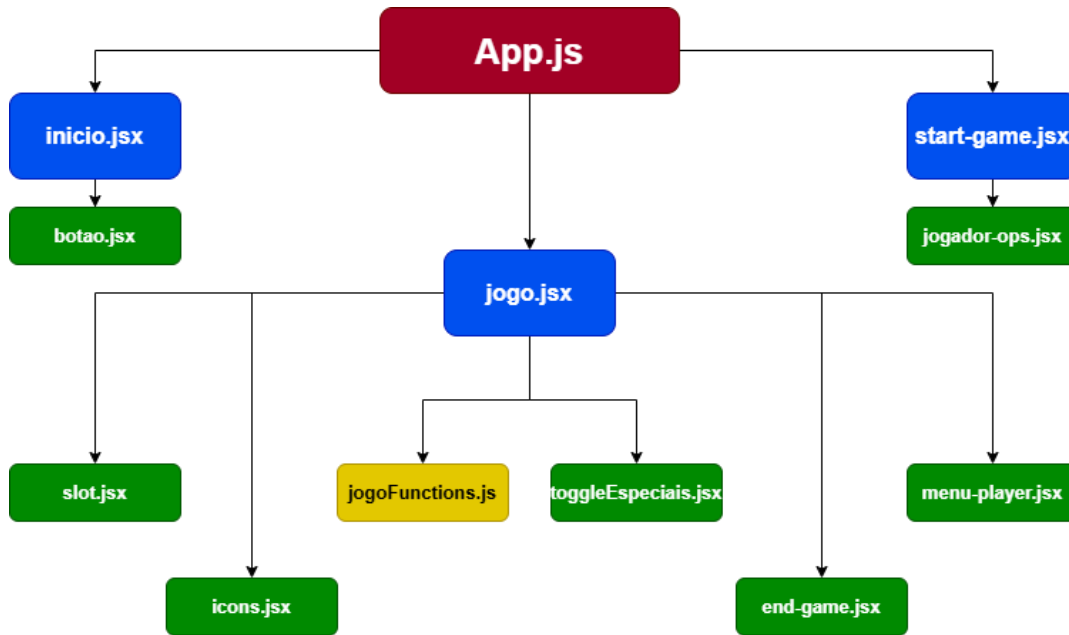


Figura 2 - Diagrama de componentes

No total o projeto dispõe de 10 componentes e 2 ficheiros (Javascript) destinado a funções externas. Sendo estes:

- **App.js:** Ficheiro que decide qual o componente a renderizar (*inicio.jsx*, *start-game.jsx* ou *jogo.jsx*) através de uma variável de estado chamada *currentMenu*;
- **inicio.jsx:** Componente responsável por mostrar o menu inicial que permite que o utilizador escolha qual o modo de jogo que pretende jogar;
- **botao.jsx:** Componente que define os botões usados no *inicio.jsx* para seleccionar o modo de jogo;
- **start-game.jsx:** Componente que apresenta o menu de pré-jogo e define as equipas a escolher e cria os jogadores;
- **jogador-ops.jsx:** Mostra as opções de pré-jogo ao utilizador, que permitem escolher a equipa e o nome;
- **jogo.jsx:** Componente mais importante da aplicação pois contém o tabuleiro, gere as jogadas com base numa matriz, verifica vitórias ou empates e trata da criação das casas especiais.

- **jogoFunctions.js:** Ficheiro que agrega funções fundamentais utilizadas em *jogo.jsx*, como verificar a existência de um vencedor, gerar casas especiais e validar se a coluna escolhida está livre para jogar.
- **slot.jsx:** Componente responsável por mostrar os discos no tabuleiro e de controlar as animações dos mesmos a cair até à casa pretendida, recorrendo a animações *CSS*, *hooks* como *useEffect* e a alguns cálculos.
- **menu-player.jsx:** Componente que exibe o nome, equipa dos jogadores e tempo disponível para cada jogada. Através de *hooks* como: *useEffect* e *useState*, é efetuado o controlo do tempo de cada jogador, assim quando um movimento é efetuado, o temporizador desse jogador é reiniciado (0 segundos), ou no caso de um jogador ao esgotar o seu tempo, perde a vez, passado para o adversário.
- **end-game.jsx:** Componente que apresenta um menu no final do jogo. Caso empate ou vença, indica o jogador vencedor e dispõe opções para voltar ao início ou reiniciar o jogo.
- **icons.jsx:** Componente que dispõe opções durante o jogo: permitindo voltar ao início e reiniciar o jogo.
- **toggleEspeciais.jsx:** Componente que permite habilitar ou desabilitar a visibilidade das casas especiais.

3. Desafios

De modo geral, o grupo não enfrentou grandes dificuldades técnicas ao longo do desenvolvimento, graças ao forte envolvimento e motivação de todos os elementos. Ainda assim, alguns desafios pontuais merecem destaque:

- **Tomada de decisões em grupo:** Definir uma linha comum de design e funcionalidade exigiu discussão e consenso sendo a diversidade de ideias positiva.
- **Animação da peça a cair:** A implementação do efeito visual em que a peça "desliza" até à posição mais baixa da coluna exigiu ajustes em termos de lógica, temporização CSS e etc...
- **Gestão do tempo por jogada:** Controlar o temporizador de 10 segundos para cada jogada, respeitando transições de turno e eventos inesperados (como tempo esgotado), implicou um bom uso do useEffect e controlo do estado para evitar comportamentos inconsistentes.
- **Modo contra o computador:** A criação de um modo de jogo contra o computador, ainda que com jogadas aleatórias conforme indicado no enunciado, exigiu lógica adicional para garantir que a jogada fosse válida e visualmente compreensível para o jogador.

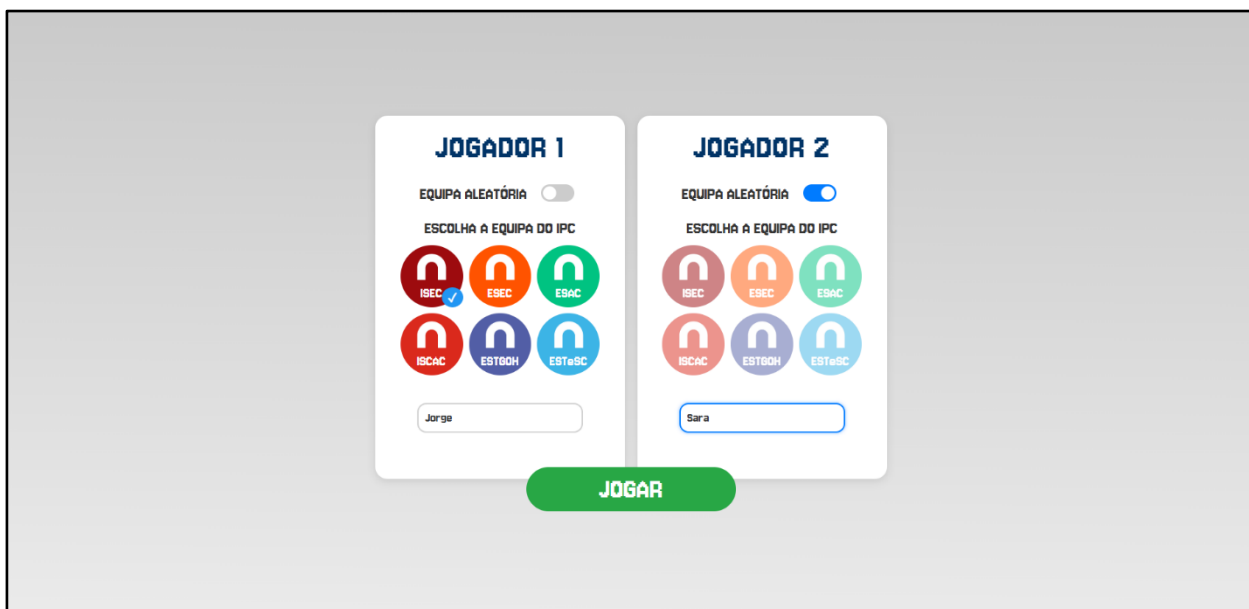


Figura 3 - Menu de Jogador v.s. Jogador

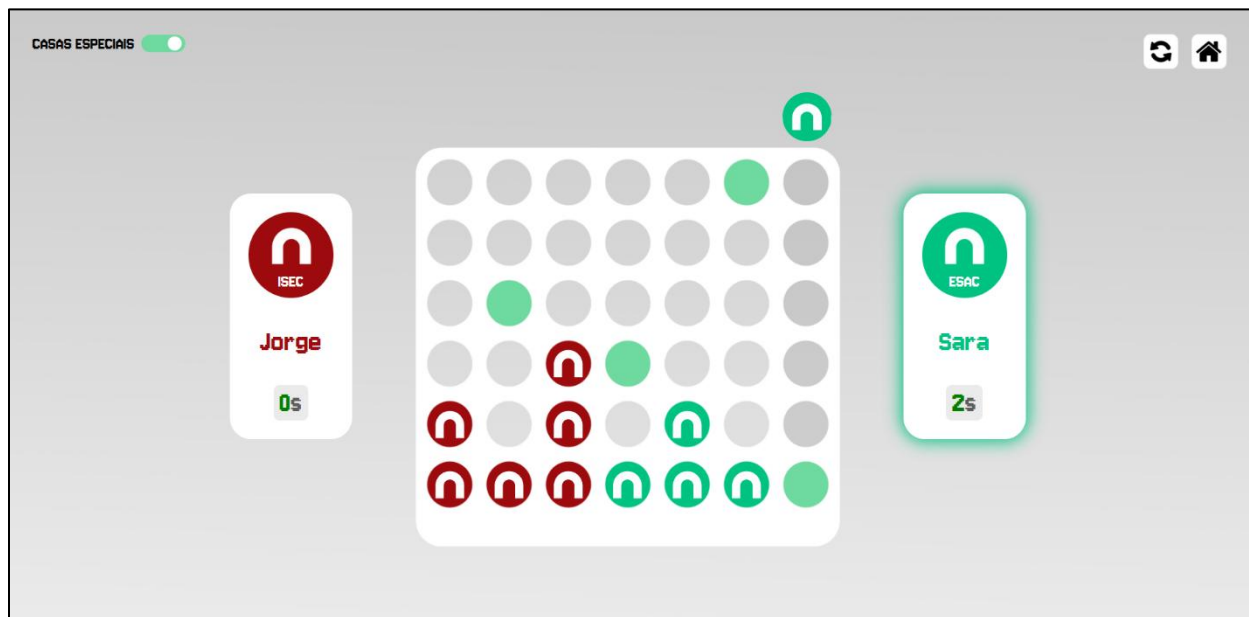


Figura 4 - Decorrer de um jogo



Figura 5 - Fim de partida por vitória