 **Departamento de Engenharia Informática e de Sistemas**

**Instituto Superior de Engenharia de Coimbra**

Instituto Politécnico de Coimbra

Licenciatura em Engenharia Informática

Curso Engenharia Informática Pós-Laboral

Unidade Curricular de Linguagem Script

Ano Letivo de 2024/2025

**Alunos:**

**Gabriel Dinis Paixão Dos Santos – 2024134573**

**Pedro Miguel Barreiro Reis – 2018013543**

**Coimbra, 30 de maio de 2025**

Relatório – 4 em Linha Especial

Conteúdo

[1. Diagrama de Componentes 2](#_Toc199643253)

[2. Funcionalidades dos Componentes 2](#_Toc199643254)

[1. App.jsx 2](#_Toc199643255)

[2. StartScreen.jsx 3](#_Toc199643256)

[3. Game.jsx 3](#_Toc199643257)

[4. ScoreBoard 3](#_Toc199643258)

[5. Board.jsx 3](#_Toc199643259)

[6. Cell.jsx 3](#_Toc199643260)

[3. Soluções Técnicas e Decisões de Desenvolvimento 3](#_Toc199643261)

[a) Gerenciamento do Tabuleiro e Células Especiais 3](#_Toc199643262)

[b) Controle de Turno e Timer 4](#_Toc199643263)

[c) Detecção de Vitória e Empate 4](#_Toc199643264)

[d) Modo Jogador vs CPU 4](#_Toc199643265)

[e) Animações e Efeitos Visuais 4](#_Toc199643266)

[f) Gerenciamento do Placar 4](#_Toc199643267)

[g) Organização dos Estilos 5](#_Toc199643268)

[Conclusão 5](#_Toc199643269)

## 1. Diagrama de Componentes

## 2. Funcionalidades dos Componentes

### App.jsx

* 1. Ponto de entrada da aplicação
  2. Gerencia o estado global: tela atual, modo de jogo, placar, nomes dos jogadores
  3. Exibe a tela inicial ou o jogo conforme o estado

### StartScreen.jsx

* 1. Permite escolher o modo (PvP ou Jogador vs CPU)
  2. Entrada dos nomes dos jogadores
  3. Validação e início do jogo com os parâmetros escolhidos

### Game.jsx

* 1. Inicializa o tabuleiro (6x7) e as células especiais (aleatórias)
  2. Gerencia o turno, timer (10s por jogador), detecção de vitória ou empate
  3. Implementa a IA (CPU joga em coluna aleatória)
  4. Exibe placar, timer, mensagens de vitória/empate/timeout
  5. Permite reiniciar ou voltar ao menu inicial

### ScoreBoard

* 1. Exibe o placar atual dos jogadores ou CPU

### Board.jsx

* 1. Exibe a grade do jogo
  2. Gerencia o hover nas colunas (efeito visual)
  3. Passa propriedades para cada célula (especial, vencedora, animação...)

### Cell.jsx

* 1. Exibe uma célula do tabuleiro
  2. Gerencia cor, efeitos especiais (especial, vencedora, queda, hover)

## 3. Soluções Técnicas e Decisões de Desenvolvimento

### a) Gerenciamento do Tabuleiro e Células Especiais

O tabuleiro é representado por um array bidimensional (6x7), onde cada célula pode ser null, 'player1', 'player2' ou 'cpu'.  
As células especiais são sorteadas aleatoriamente no início de cada partida e armazenadas em um conjunto para garantir que não se repitam.

### b) Controle de Turno e Timer

O turno é alternado entre os jogadores após cada jogada válida.  
Um timer de 10 segundos é implementado usando setInterval e useRef para garantir que o tempo seja controlado corretamente mesmo com re-renderizações do React.  
Se o tempo do jogador acabar, o turno é passado automaticamente e uma mensagem é exibida.

### c) Detecção de Vitória e Empate

Após cada jogada, o sistema verifica se há quatro peças do mesmo jogador alinhadas em qualquer direção (horizontal, vertical ou diagonal).  
Se houver, as coordenadas das peças vencedoras são armazenadas para animação.  
Se o tabuleiro estiver cheio e não houver vencedor, o jogo é declarado empatado.

### d) Modo Jogador vs CPU

No modo contra o computador, a IA faz uma jogada simples: escolhe aleatoriamente uma coluna válida e posiciona a peça.  
A jogada do CPU ocorre automaticamente após o turno do jogador, com um pequeno atraso para simular "pensamento".

### e) Animações e Efeitos Visuais

Animações CSS são usadas para a queda das peças, destaque das peças vencedoras e indicação de células especiais.  
O hover nas colunas do tabuleiro indica visualmente onde a peça será posicionada.  
Células especiais são destacadas com uma borda ou cor diferenciada.

### f) Gerenciamento do Placar

O placar é controlado pelo componente principal (App) e atualizado sempre que um jogador vence uma partida.  
O componente Game, ao detectar uma vitória, comunica ao App qual jogador venceu, e o placar é incrementado para esse jogador.  
O ScoreBoard recebe o placar atualizado via props e exibe em tempo real na interface.  
O placar é mantido enquanto o usuário permanecer na aplicação, sendo reiniciado apenas se o usuário voltar à tela inicial ou recarregar a página.

### g) Organização dos Estilos

Os estilos CSS são organizados por componente, facilitando a manutenção e a personalização visual do projeto.  
Utilização de classes para diferentes estados das células (normal, especial, vencedora, animada).

## Conclusão

O projeto "4 em Linha Especial" foi desenvolvido com foco em modularidade, clareza de interface e experiência do usuário.  
A arquitetura baseada em componentes React permite fácil manutenção e expansão futura, enquanto as soluções técnicas adotadas garantem robustez e interatividade ao jogo.