

## Trabalho Prático em React

### “4 em linha Especial”

#### > **Âmbito**

Este trabalho prático (TP) tem como objetivo o desenvolvimento de uma aplicação em **React JS**, de forma a que os alunos possam aplicar os conhecimentos adquiridos nas aulas. Como tal, devem demonstrar o domínio em **React** e nas tecnologias necessárias ao desenvolvimento da aplicação, nomeadamente a linguagem JavaScript, HTML e CSS.

Todos os detalhes do TP encontram-se especificados nas secções seguintes e, como tal, **sugere-se a leitura com a devida atenção** sob pena de não implementarem o trabalho na sua totalidade. Recorda-se que o TP tem um peso de **8 valores** da nota final da disciplina, como especificado na FUC de Linguagens Script.

Valoriza-se a autonomia do aluno na construção de uma solução para o tema proposto, tendo em consideração que esta deve cumprir, no mínimo, os requisitos gerais apresentados. A **submissão de trabalhos da autoria de terceiros, implica, para além de 0 na nota final, a ativação dos devidos processos legais de plágio, aplicados no ISEC/IPC.**

#### Considerações:

- Os alunos devem organizar-se **em grupos de 2 ou 3 alunos**. **Não são permitidos trabalhos individuais.**
- A formalização do grupo deve ser efetuada **obrigatoriamente** no formulário disponível no *inforestudante*, secção “**Submissão de Trabalhos**” – **Grupo de Trabalho de LS**, **até ao dia 17 de maio**. Apenas **um dos elementos necessita especificar o grupo**, devendo ainda **associar os colegas** (conforme figura abaixo). Na formalização do grupo, devem submeter o documento disponibilizado e preenchido. **Todos os alunos do grupo** devem colaborar na implementação do trabalho, sob pena de, na defesa, **lhes ser atribuído 0 na nota do trabalho ou serem removidos do grupo inicialmente especificado, no momento da entrega.**
- Os alunos que pretendam **manter a nota do TP obtida em 2023/2024**, devem submeter os dados no formulário, secção “**Submissão de Trabalhos**”, cujo título “**Manter nota do TP de LS**”, **até ao dia 17 de maio**. Apenas é possível manter notas superiores ou iguais a **50%**.
- O não cumprimento das regras detalhadas no enunciado, implica penalização na nota do TP.



Nome	Número	Curso	E-mail
XXXXXXXXXXXX	XXXXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX@isec.pt

## > Tema

O tema do TP é uma **variante** do tradicional jogo “**4 em Linha**”, no qual devem ser efetuadas jogadas alternadas entre dois jogadores ou entre um jogador e o computador, até se encontrar um vencedor ou um empate.

O **jogo** é constituído por uma grelha 6x7 (linha x coluna) e termina quando um jogador se torna vencedor ao conseguir efetuar 4 em linha (seja na horizontal, vertical ou diagonal). Para isso, o jogador deve selecionar, com o rato, a coluna no qual pretende inserir a peça, num determinado tempo, caso contrário perde a sua vez. Para além deste tempo limite para cada jogada, devem existir **células especiais** que dão bónus ao jogador em jogo quando as seleciona. Estas características estão especificadas na secção seguinte.

As figuras seguintes apresentam exemplos do jogo, não estando nessas figuras todas as opções e funcionalidades referidas na secção seguinte.

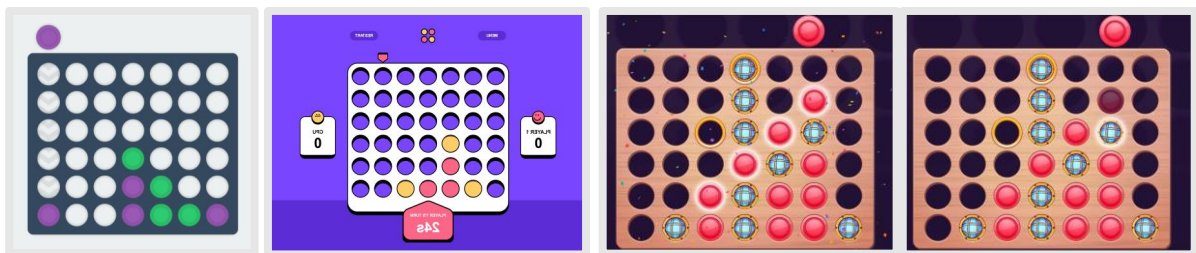


Figura 1 – Exemplos do Jogo 4 em linha

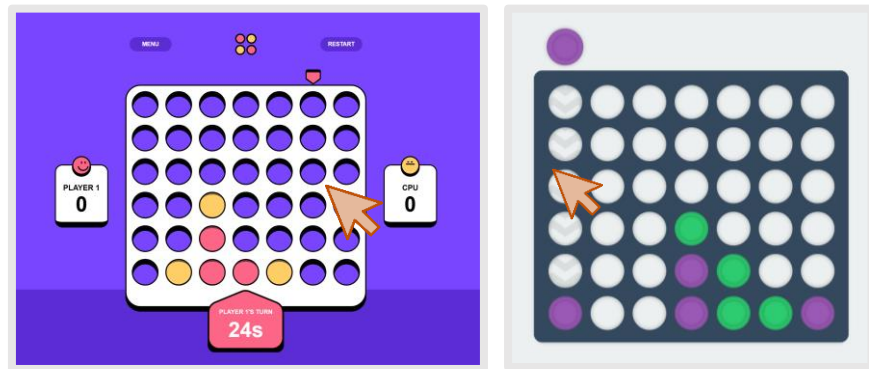
Fontes: <https://www.crazygames.com/game/4-in-a-row-connected-multiplayer-online>  
<https://www.coolmathgames.com/0-four-in-a-row>

## > Funcionalidades

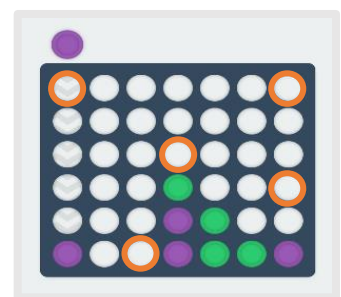
O jogo deverá ser implementado em **React**, e disponibilizar, pelo menos, as seguintes **funcionalidades**:

- Solicitar o nome dos jogadores;
- Decidir de forma aleatória o primeiro jogador bem como a cor da peça, e apresentar essa informação;
- Apresentação do tabuleiro geral de jogo;
- Identificação clara do jogador que deve efetuar a jogada, seja por cor ou outra forma que o aluno desejar;
- Apresentar tempo disponível, que deverá iniciar, em contagem crescente, sempre para cada jogada. O tempo máximo para cada jogada deverá ser 10 segundos. Caso o jogador não jogue, passa a sua vez;

- Selecionar o espaço onde deve ser colocada a peça do jogador:
- **Opção 1:** O jogador **seleciona exatamente** o local onde a peça deve ser colocada, com um clique no local pretendido, sabendo que, apenas devem ser aceites os cliques das últimas células livres dessa coluna;
  - **Opção 2:** É apresentado no topo da coluna, fora do tabuleiro, um elemento, seja uma seta ou a própria peça (como se mostra nas figuras seguintes), para identificar a coluna onde o rato se encontra. Isso acontece enquanto o jogador navegar no tabuleiro, devendo o elemento mover-se também no topo da coluna, e a coluna em questão ser destacada. Quando houver um clique, a peça aparece instantaneamente na posição a ser inserida (última célula livre dessa coluna);
  - **Opção 3 (mais valorizada):** Implementa a opção 2, mas a peça desliza pela coluna, até à posição onde deve ficar



- Gerar, de forma aleatória, 5 espaços especiais, que quando usados/selecionados, permitem que o jogador que a ativou jogue novamente, antes de passar a vez ao outro jogador, portanto, ao selecionar essa célula, o jogador é o mesmo para a próxima jogada. Estas células especiais são fixas durante um jogo e devem estar de alguma forma destacadas, como por exemplo, um bordo/fundo diferente;
- **2 níveis de jogo** em que a forma de jogar varia de acordo com os jogadores em jogo, nomeadamente: **1 contra 1 jogador** e **1 contra computador** (Não se pretende um algoritmo elaborado, apenas um valor aleatório nas células disponíveis);
- Identificação de fim de jogo, quando for efetuado 4 em linha e existir um vencedor, ou não existirem mais espaços para jogar e resultar num empate;
- Permitir jogar novamente;



## > **Implementação**

A implementação do trabalho, como anteriormente referido, **deverá ser efetuado em React**, recorrendo, pelo menos, à versão 18. Em relação à implementação, é dada a liberdade ao aluno para seguir a estratégia que considerar melhor, no entanto, especificam-se abaixo **algumas características e limitações** que devem ter em consideração.

**Relembra-se que o React DOM** é o responsável por atualizar o DOM e apresentar os elementos React. Assim, alerta-se para a **não utilização de métodos** como **getElementById** ou o **querySelector**, para acesso e manipulação direto dos elementos DOM do browser, apenas para o nó raiz. Além disso, os alunos não devem efetuar a gestão dos eventos com recurso ao **addEventListener**, mas sim usando o sistema de gestão de eventos do React.

### → **Criação do Projeto**

Os alunos **devem criar um projeto** de forma a facilitar a gestão de dependências de todos os módulos necessários à implementação de uma aplicação React, recorrendo à aplicação **create-react-app** ou usando a base do projeto fornecido nas fichas práticas.

### → **Componentes**

Em relação à criação dos componentes, o React permite a implementação dos componentes usando abordagens diferentes, no entanto, **o trabalho deverá ser implementado exclusivamente com componentes funcionais**.

### → **Interface**

Fica à responsabilidade do aluno a especificação da interface da aplicação. Esta não tem de ter o aspeto do exemplo apresentado nas figuras anteriores. Deve haver uma preocupação no sentido de desenvolver uma aplicação visualmente agradável com uma imagem consistente. Poderão ser usadas bibliotecas e *frameworks* CSS que considerem úteis e possam facilitar a implementação da interface, como por exemplo, *Bootstrap*, *Tailwind CSS*, entre outros. No entanto, **não é permitido** recorrer a bibliotecas e/ou frameworks que disponibilizem componentes React já implementados, como por exemplo, *Material UI*, *React Bootstrap*, *Semantic UI*, *React Toolbox*, *Ant Design*, *React Foundations*, entre outros. **Todos os componentes necessários à implementação do jogo devem ser implementados pelos alunos. Se assim o entenderem, podem reutilizar algum dos componentes implementados nas aulas práticas.**

Nesse sentido, sempre que o aluno utilize componentes previamente implementados por terceiros (bibliotecas/frameworks, entre outros), será considerado que o aluno não implementou esse componente nem as respectivas funcionalidades.

### > **Relatório**

O TP deve ser acompanhado de um pequeno relatório técnico, com o máximo de 3 páginas, onde devem **apresentar obrigatoriamente o diagrama de componentes**, as funcionalidades dos componentes implementados e as soluções utilizadas no desenvolvimento da aplicação.

### > **Condições e Data de Entrega**

O TP deverá ser submetido até às **8h** do dia **2 de junho de 2025**. A entrega é realizada em formato digital, **no inforestudante, apenas por um elemento do grupo**, devendo indicar no momento da submissão os outros elementos do grupo de trabalho. O TP prático deve ser submetido num ficheiro **ZIP** com o nome **LS2425\_num1\_num2\_num3.ZIP**. **Trabalhos práticos submetidos depois do horário terão uma penalização de 5% da nota final, por cada hora de atraso. Não se aceitam trabalhos práticos enviados por email nem após as 23h59m do dia 2.**

**NOTA IMPORTANTE:** Antes de criar o ficheiro ZIP, os alunos devem **obrigatoriamente** remover a pasta **"node\_modules"** e o ficheiro **"package-lock.json"** existente na estrutura do projeto. Para além do projeto implementado, que deve incluir o código fonte (sem a referida pasta), o ficheiro Zip deve incluir o relatório em formato pdf.

Recomenda-se que os alunos testem o trabalho que submetem, sob pena, de não entregarem a versão correta ou pretendida, como frequentemente acontece....

### > **Defesa**

O trabalho terá uma defesa obrigatória, que será efetuada **em data a definir**, através de uma pré-inscrição no *inforestudante*. **A não comparência** à defesa na data (dia e hora) combinada, implica a anulação do trabalho e consequente reprovação do aluno à disciplina.

### > **CrITÉrios de Avaliação**

- Aspeto Gráfico (10%);
- Solicitar e apresentar o nome dos jogadores, destacando o jogador em cada jogada. O primeiro jogador a ser obtido de forma aleatória (10%);
- Apresentar o tabuleiro de jogo (2.5%);

- Seleção do espaço/célula onde deve ser colocada a peça do jogador (**15%** na opção 1, **25%** na opção 2 e **30%** na opção 3):
  - **(opção 1)** Jogador seleciona especificamente o espaço vazio onde a peça deverá ser colocada, não devendo aceitar local que não seja o correto;
  - **(opção 2)** Jogador movimenta o rato no tabuleiro e a coluna onde o rato se encontra, é destacada. Ao existir o clique, a peça é colocada diretamente no espaço correto;
  - **(opção 3 – mais valorizada)** Jogador movimenta o rato no tabuleiro e a coluna onde o rato se encontra, é destacada. Ao existir o clique, a peça desliza até ao local correto;
- Apresentar aleatoriamente espaços especiais no tabuleiro e permitir jogar novamente quando uma peça é colocada nesse espaço. Isto é, quando uma peça atinge um espaço especial, o jogador passa a efetuar nova jogada, e só depois, é que passa a vez (**15%**);
- Apresentar tempo de cada jogada (**5%**)
- Permite 2 níveis, isto é, permite jogar com computador, sendo que deverá ser dado algum tempo para mostrar onde efetuou a jogada (**15%**)
- Identificação de fim de jogo/vencedor – seja porque encontrou 4 em linha ou porque não existem mais espaços disponíveis para jogar (**10%**)
- Permitir jogar novamente (**2.5%**)
- ***Critérios gerais a aplicar à nota:***
  - Organização do código e metodologia usada na implementação da aplicação, nomeadamente, forma de estruturação da aplicação em vários componentes React, o uso correto de *hooks* (quando aplicável) na aplicação, entre outros.
  - Defesa do aluno.