

Introdução às Tecnologias Web

Enunciado do Projeto de 2022/2023

Tema: Bejeweled

Em 2001, a PopCap Games lançou um jogo de combinação de peças chamado Bejeweled que, até 2008, foi descarregado cerca de 150 milhões de vezes¹ e que ainda hoje é muito popular, por exemplo em telemóveis. O objetivo é alinhar três ou mais joias iguais, fazendo com que sejam eliminadas de um tabuleiro retangular, causando ocasionalmente uma reação em cadeia em que as joias adjacentes, ao ocuparem as casas que ficaram vagas, geram outros alinhamentos e eliminações de joias. Esta é a ideia base, sendo mais à frente detalhados os requisitos essenciais que devem ser cumpridos, bem como funcionalidades adicionais que os grupos podem explorar para ter melhor nota. Mais informações sobre o jogo podem ser consultadas na Wikipedia em <https://en.wikipedia.org/wiki/Bejeweled>.

Objetivos

1. Criação de um *site* com **informação sobre o jogo**, incluindo: a) conteúdos de *texto*, tais como regras, tutoriais, curiosidades históricas, listas de opções, lista de campos para o registo de jogadores, tabelas com pontuações e estatísticas, e ficha técnica com os autores e tecnologias usadas; b) conteúdos *gráficos*, como imagens e animações dos avatares dos jogadores, dos personagens ou objetos que participam ativamente no jogo, e dos elementos decorativos do tabuleiro, bem como vídeos, por exemplo de demonstração de funcionalidades; e c) conteúdos *áudio*, como música e sons para ilustrar as ações durante o jogo.

Todos estes conteúdos devem fazer parte de páginas HTML, ligadas entre si de forma coerente, isto é, sem que o jogador se perca durante a aprendizagem e uso do jogo. A apresentação dos conteúdos deve ser especificada usando CSS.

2. Desenvolvimento das **funcionalidades do jogo**, nomeadamente: inicialização do jogo e do tabuleiro, jogadas permitidas a cada jogador em cada momento, deteção do fim do jogo, cálculo de pontuações e estatísticas, e registo de jogadores, eventualmente suportando vários modos com regras distintas.

A programação das funcionalidades deve ser feita em JavaScript, e, no final, deve haver uma integração total com o *site* do objetivo 1.

¹ Mark Ward (2008). Casual games make a serious impact. *BBC News*. <http://news.bbc.co.uk/2/hi/technology/7301374.stm>.

Requisitos essenciais do jogo

Os requisitos seguintes devem ser suportados pelos jogos de *todos* os grupos, servindo de base para a avaliação, e valem **14 valores** da nota no projeto.

1. O jogo tem de suportar o *modo individual*, isto é, com apenas um jogador do princípio ao fim de um jogo, não sendo preciso passar a vez a outro jogador.
2. Antes de poder jogar, um jogador tem de se *identificar*, podendo ser necessário registar-se (*e-mail*, senha, faixa etária, e género), caso ainda não o tenha feito. Os registos de dados devem ficar guardados em *local storage* do *browser*.
3. De igual forma, só depois de se identificar é que um jogador pode ver *estatísticas* sobre o seu número de jogos, tempo total jogado, e pontuação máxima.
4. No início de cada jogo, é mostrado um *tabuleiro* de dimensão 8×8 , preenchido com joias de sete tipos diferentes, dispostas aleatoriamente, não podendo estar alinhadas três joias do mesmo tipo na horizontal.
5. Durante um jogo são mostrados o *tempo decorrido*, em segundos, a pontuação, o número de joias eliminadas até ao momento e quantas têm de ser eliminadas para o jogo terminar, bem como um botão para terminar manualmente o jogo.
6. Uma *jogada* consiste em selecionar duas joias adjacentes, na horizontal ou na vertical, sendo estas trocadas (uma ocupa a casa da outra e vice versa) se, em consequência disso, três ou mais joias iguais ficarem alinhadas na horizontal.²
7. Quando três ou mais joias iguais ficam alinhadas na horizontal, são *eliminadas* do tabuleiro, sendo que as joias acima das eliminadas descem no tabuleiro para ocuparem as casas que ficaram vagas. Ao mesmo tempo, no topo do tabuleiro surgem novas joias escolhidas aleatoriamente.
8. Pode acontecer uma *reação em cadeia* se, após a eliminação e reposicionamento de joias, o tabuleiro ficar com três ou mais joias alinhadas na horizontal, sendo necessário repetir os passos do requisito 7.
9. Os *pontos* obtidos por alinhamento de n joias iguais, em que $n \geq 3$, são calculados da seguinte forma: $(n - 3) + 1$. Por exemplo, se forem alinhadas três joias iguais, o jogador ganha um ponto.
10. A *seleção de joias* pode ser feita com o rato ou com o teclado, sendo suficiente apenas um destes dispositivos de entrada. Por exemplo, com o rato bastaria fazer clique sobre uma joia para a selecionar.

² Não é necessário verificar alinhamentos verticais de joias na versão essencial do jogo.

11. Podem existir tabuleiros em que, por causa da disposição das joias, seja *impossível fazer uma jogada*, podendo o jogador terminar manualmente o jogo carregando no botão mencionado no requisito 5.
12. Um jogo *termina* quando for atingido o objetivo de eliminar vinte ou mais joias do tabuleiro, ou se for carregado o botão de terminar manualmente o jogo.
13. Após um jogo em que o objetivo é atingido, deve ser apresentada uma *tabela com as dez melhores pontuações* e respetivos tempos de jogo (que podem ser usados para desempatar) e os nomes dos jogadores, devendo essa informação ficar guardada em *local storage* do *browser*.
14. Os conteúdos nas páginas do *site*, e o jogo propriamente dito, devem *ajustar-se automaticamente* ao tamanho e orientação do ecrã (*responsive web design*).

Funcionalidades adicionais do jogo

Os **6 valores** restantes da nota do projeto provêm de funcionalidades adicionais que tornem o jogo mais desafiante e apelativo. Apresentam-se de seguida algumas sugestões, que não têm todas de ser concretizadas, sendo incentivada a criatividade para encontrar e implementar ideias novas.

- Verificação de *alinhamentos verticais* de joias, para além dos alinhamentos horizontais mencionados no requisito 7, mantendo-se o mínimo de três joias iguais para serem eliminadas do tabuleiro bem como a lógica de ocupação das casas que ficaram vagas e a colocação de novas joias no tabuleiro.
- *Deteção automática da impossibilidade de fazer jogadas*, de acordo com a definição de jogada no requisito 6. Se tal suceder, o jogador deve ser informado e deve aparecer um botão que serve para baralhar as joias no tabuleiro e que permite continuar a jogar, tendo associada uma penalização de cinco pontos.
- *Botão de dica*, que mostra um par de joias no tabuleiro (se existir) em que é possível fazer uma jogada, tendo, contudo, uma penalização de um ponto.³
- *Várias configurações de tabuleiros*, com dimensões e número de joias a eliminar à escolha do jogador, podendo existir níveis de dificuldade predefinidos, tais como: fácil, com 8×8 casas e 20 joias a eliminar, intermédio, com 9×9 e 25 joias a eliminar, e difícil, com 10×10 e 30 joias a eliminar. Cada configuração de tabuleiro deve ter a sua tabela de pontuações, como no requisito 13.

³ Se as dicas estiverem a funcionar corretamente, bem como a possibilidade de baralhar as joias do tabuleiro caso seja impossível fazer uma jogada, o jogo pode ter um *modo de demonstração*, em que o computador é que faz as jogadas automaticamente.

- *Temas para tabuleiros*, com diferentes imagens de fundo, símbolos gráficos para as joias (que podem até deixar de ser joias e passar a ser outro tipo de objetos), animações, músicas, e sons. Também podem existir joias especiais que, quando eliminadas do tabuleiro, permitem obter mais pontos, por exemplo, causando a eliminação de todas as joias na vertical e/ou na horizontal.
- *Outros modos de jogo*, com outros objetivos (para além do mencionado no requisito 12), tais como: fazer o máximo de pontos até ser atingido um limite de jogadas ou de tempo,⁴ ou eliminar um mínimo de joias de cada tipo. Cada modo de jogo deve ter a sua tabela de pontuações, como no requisito 13.
- *Modo multijogador*, em que dois (até quatro) tabuleiros com a mesma configuração são mostrados lado-a-lado, para permitir que os jogadores vão pensando nas suas próximas jogadas enquanto aguardam a vez. O jogador que estiver a jogar tem um tempo limite durante o qual pode fazer todas as jogadas que conseguir. Os jogadores têm de se inscrever no jogo antes de este começar.⁵ O jogo termina quando algum jogador tiver atingido o objetivo estipulado, como, por exemplo, o que vem mencionado no requisito 12.
- *Segundo dispositivo de entrada*, nomeadamente, dos dois indicados no requisito 10, o que não tiver sido suportado inicialmente.

A ter em conta

- No início de todos os ficheiros de código (HTML, CSS, e JavaScript) deve estar, em comentário, a sigla de unidade curricular, o ano letivo, o número do grupo, e o nome, número, e turma PL dos alunos.
- O código JavaScript deve estar comentado, as variáveis devem ter *nomes inteligíveis*, e devem ser respeitadas regras de alinhamento dos blocos de código.
- Os ficheiros devem estar organizados em pastas, sendo sugeridas as seguintes: *images* ou *media* (para imagens, vídeos, e sons), *styles* (para ficheiros CSS), e *scripts* (para ficheiros JavaScript), ou o equivalente em português.⁶

⁴ No modo de jogo com limite de tempo, pode existir um botão de pausa para o jogador poder descansar, ficando o tabuleiro escondido.

⁵ A inscrição dos jogadores deve ser feita numa página que antecede a do jogo. Nessa página, do lado esquerdo pode aparecer uma lista com os jogadores inscritos (inicialmente vazia) e do lado direito pode estar um formulário para cada jogador, à vez, introduzir o seu *e-mail* e senha. Quando todos os jogadores se tiverem inscrito, podem então avançar para a página do jogo.

⁶ MDN Web Docs (2023). *Getting started with the web: Dealing with files*. https://developer.mozilla.org/docs/Learn/Getting_started_with_the_web/Dealing_with_files.

- É bem conhecido da indústria do *software* que é gasto mais esforço a corrigir e atualizar código existente do que a produzir código novo.⁷ Apesar de poderem existir jogos semelhantes na Internet, quem se inspirar nos mesmos deve ter em conta que o esforço de interpretação do código e de adaptação das funcionalidades pode não compensar, especialmente se for reconhecido que a aprendizagem passa muito pela experiência de aplicação de conhecimentos.

Entregas

- Cada objetivo do projeto tem uma entrega associada, realizada através de uma atividade na página de entrada do Moodle da unidade curricular.
- Para o objetivo 1, a cumprir até ao dia **9 de abril** (domingo), deve ser entregue um ficheiro com nome ITW-GXX-E1.ZIP, onde XX representa o número do grupo, com dois dígitos (ex. 01).
- Para o objetivo 2, realizável até ao dia **21 de maio** (domingo), o ficheiro deve chamar-se ITW-GXX-E2.ZIP.

Bom trabalho no projeto!

⁷ Ver páginas 1 a 3, sobre a natureza do *software*, no livro Roger S. Pressman e Bruce R. Maxim (2015). *Software engineering: A practitioner's approach* (8ª edição). McGraw-Hill.