

Modelo de Cenário de Aprendizagem



Disciplina: Programação e Sistemas Informação (PRSI)
Módulo/ Unidade didática: M3 - Programação Estruturada
Turma: 10º GPSI
Autor: Filipe Marques

Breve descrição

Em que disciplina e respetiva temática se inscreve este cenário? De que modo este contribui para o desenvolvimento das competências preconizadas na disciplina?

Este cenário desenrola-se numa disciplina introdutória à programação para um curso profissional. A índole mais profissionalizante obriga aqui a um trabalho acrescido de reforço dos fundamentos da programação, regras de boa escrita, correcção algorítmica e principalmente, neste módulo, a capacidade de abstração. Tudo isto, são competências fundamentais na área da programação, que não sendo trabalhadas nesta fase, poderão dificultar o avanço do aluno rumo à resolução de problemas mais complexos, que exigem sempre uma maior estruturação do pensamento.

Com este fim, os alunos vão desenvolver um jogo sob a forma de um corpo de funções/procedimentos que permitam o seu controlo via consola. A representação visual do jogo será puramente textual, e terá Loops/Ciclos para controlar a sua renderização. Será também uma oportunidade para os alunos perceberem a mecânica por detrás da criação de um jogo. Ao longo do projeto os alunos serão incentivados para a utilização de funções/procedimentos, promovendo uma boa organização do código, a sua modularidade, a generalização e reutilização.

Liskov and Guttag (2000) destacou 3 tipos de abstração: abstração procedimental, abstração de dados, e abstração iterativa.

Segundo Wing (2006), a capacidade de conceptualização, da qual faz parte o pensamento computacional, requer da nossa parte múltiplos níveis de abstração.

A abstração procedimental é o ato de usar funções ou procedimentos para encapsular um pedaço de código que executa uma determinada tarefa, de forma bem específica, de forma a que outro código invoque o mesmo, sem haver necessidade de perceber como é que essa tarefa foi implementada. Ou seja, uma boa capacidade de abstração procedimental exige ao aluno a capacidade de encarar cada função/procedimento como uma “caixa-negra” que resolve um problema bem definido, independentemente do ambiente/contexto em que é executada.

Objetivos de Aprendizagem

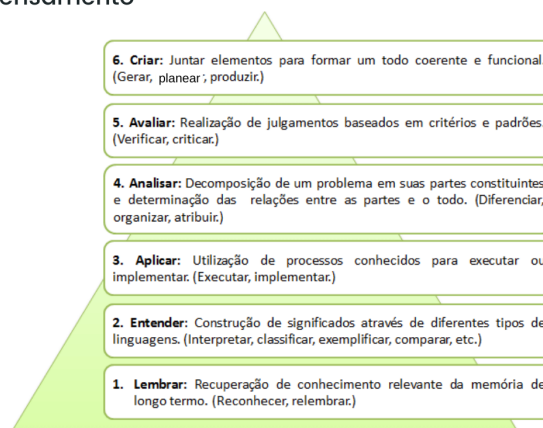
Quais os objetivos de aprendizagem assumidos para este cenário? Como se relacionam com as competências que os alunos da turma deverão desenvolver?

O principal objetivo será a promoção de uma melhor organização do código escrito, estruturando o algoritmo e o pensamento. Este objetivo trabalha diretamente com os níveis mais a topo da taxonomia de Bloom de acordo com a versão revista da taxonomia por Anderson (2000):

- Analisar e decompor um problema nas suas partes constituintes e estabelecer uma relação entre as partes e o todo.
- Planear e criar elementos que formam um todo coerente e funcional.

Objetivo

Promover uma melhor organização do código escrito, estruturando o algoritmo e o pensamento



Os objetivos específicos:

- Declarar variáveis locais e globais
- Entender o ciclo de vida e *scope* de uma variável
- Segmentar um programa em funções e procedimentos
- Utilizar passagem por parâmetros nas funções
- Utilizar Ciclos e condicionais

Estes objetivos que se pretendem atingir com este cenário, tendo em vista que será trabalhado em contexto da metodologia PRIMM, permitem trabalhar as diversas competências previstas no Perfil do Aluno à saída da escolaridade obrigatória.

Papel dos Alunos

Em que tipo de atividades serão envolvidos os alunos?

A partir de um guião fornecido, os alunos irão envolver-se no desenvolvimento passo-a-passo de um jogo, procurando pensar não só na algoritmia necessária ao seu desenvolvimento, mas também nas questões de organização do código e boa escrita do mesmo. Os alunos vão, pela prática, dar-se conta da importância de segmentar e reduzir o código em componentes que possuam finalidade e comportamento bem definidos.

Que tipo de competências Séc. XXI irão essas atividades promover nos alunos?

Serão trabalhados os 4Cs a saber, a Comunicação, Colaboração, Criatividade e pensamento Crítico.

Comunicação: através de estratégias que envolvam comunicação entre pares;

Colaboração: onde se focam capacidades desenvolvidas através da interação, discussão, diálogo e partilha;

Criatividade: recorrendo a atividades colaborativas que promovam o desenvolvimento de pensamento crítico, fundamental para a reflexão e para a resolução de problemas;

Pensamento crítico, ligado à capacidade de pensar e refletir sobre as diferentes situações, sendo essencial para a resolução de problemas.

Papel do Professor

Que deve fazer o professor para orientar a aprendizagem e assegurar que os alunos alcancem os seus objetivos?

O professor dará o mote para o tema do projeto, objetivos e critérios de avaliação, acompanhado por um guião passo a passo que será disponibilizado aos alunos. O professor acompanhará todo o desenvolvimento, grupo a grupo e orientará os alunos de forma a atingirem os objetivos pretendidos. Deverá adoptar estratégias que motivem o aluno a envolver-se na sua própria aprendizagem e lhe permitam desenvolver a sua autonomia e iniciativa.

O professor não deverá perder de vista o principal objetivo deste módulo, stressando a importância de uma boa organização do código, estruturação do algoritmo, promovendo a sua modularidade e reutilização. Deverá levar o aluno a entender que certas partes do código se repetem, e por uma questão de optimização do código e economia de tempo, deverá segmentar o código em blocos funcionais compactos e bem definidos. Será importante realçar a importância de esconder certos detalhes inúteis do código, substituindo-os sob a forma de procedimentos.

Desta forma o aluno trabalhará a sua capacidade de abstracção e generalização, enfim, algumas das principais valências quando se trabalha na área da computação.

Que tipo de competências irá estas atividades promover em mim enquanto docente de acordo com o UNESCO ICT competency framework for teachers ou considerando o DigCompEdu?

São múltiplas as competências desenvolvidas no âmbito da aplicação deste cenário, em cada um dos seguintes domínios:

Currículo: não só de Informática, como de Física e Química e de Matemática, com potencial de alargamento para um contexto STEAM;

Pedagogia: pelo desenvolvimento da capacidade de resolução de problemas complexos;

Competências digitais: pela utilização de recursos digitais inovadores, amplos e pertinentes;

Organização e administração: trabalho colaborativo potencial devido ao cruzamento de componentes curriculares múltiplas;

Finalmente, pelo modelo sistematizado PRIMM de ensino adotado passível de facilmente ser transposto para outros contextos de iniciação à programação com sucesso.

Ferramentas e Recursos

Que recursos, inclusive tecnológicos, será pertinente usar? De que modo serão usados?

Computadores sala ou portátil pessoal

Ligação à Internet

Visual Studio

Videoprojetor

Quadro Branco

Guião de Projeto

[Link para Plataforma com os Recursos Educativos Digitais necessários para este cenário](#)

Pessoas e lugares

Quem mais estará envolvido no cenário (outros docentes, membros da comunidade, empregadores, especialistas externos, etc.) e que papel desempenhará cada um deles? Considere papéis não tradicionais.

Onde terá lugar a aprendizagem: na sala de aula, na biblioteca, ao ar livre, num ambiente online?

Neste cenário, a aprendizagem está centrada no aluno, tendo aqui o papel principal, como agente que assume a sua aprendizagem, desenvolvendo a sua autonomia e capacidade de aprender a aprender.

A implementação deste cenário terá lugar em ambiente de sala de aula, mas também poderia ser implementado em modalidade de e-learning, uma vez que todos os recursos necessários estão disponíveis na plataforma. Nesse contexto, seria útil o estipular de *milestones* bem definidas, onde após o avanço em cada uma dessas principais etapas, haveria lugar a sessões para acompanhar o trabalho desenvolvido, mas onde o grosso do trabalho será sempre desenvolvido autonomamente, tal como num cenário em *flipped classroom*.

Metodologias de Aprendizagem

Que metodologias de aprendizagem e estratégias de ensino serão adotadas? Qual a sua ligação às atividades, aos objetivos e à avaliação?

A metodologia base será o PRIMM (Predict, Run, Investigate, Modify, Make). O PRIMM é um método de ensino de programação, que se baseia no conhecimento de programação, tendo como ponto de partida o funcionamento do programa antes de começarem a editar e escrever os seus próprios programas. A sigla PRIMM, significa Predict (prever); Run (executar); Investigate (investigar); Modify (modificar); Make (fazer); (Sentance, Waite e Kallia, 2019). Num estudo realizado por Sentance, Waite e Kallia (2019), estes consideram que a aplicação em aula do método PRIMM teve um impacto

O PRIMM será trabalhado em Peer programming, onde duas pessoas olham para o código, podendo uma escrever o código e outro analisar o mesmo, podendo depois trocar papéis.

A Gamificação, enquanto processo que será incorporado na prática educativa para este cenário, de forma a promover a motivação e a aprendizagem de conceitos mais complexos como a abstração. Para esse efeito, serão realizados mini-quizzes ao final de cada sessão de forma a testar se os conceito(s) chave foram efetivamente assimilados.

Tempos

12h previstas para implementação deste cenário.

Avaliação

Como as atividades desenvolvidas serão avaliadas (tipo de avaliação, instrumentos, ...)? Sobre o que se foca (objetivos, competências, ...)?

Tal como estipulado no referencial desta disciplina, os procedimentos de avaliação dos alunos decorrem da natureza eminentemente prática e experimental da disciplina, privilegiando-se a vertente formativa da avaliação, indispensável à orientação do processo de ensino/aprendizagem e à auto-regulação do aluno.

As aprendizagens dos conceitos chave desenvolvidos em cada etapa do guião, em cada sessão, serão validadas por meio de exit tickets, isto é, mini quizzes.

Sumativamente, o projeto final do jogo realizado em grupo será submetido para avaliação a partir da plataforma de recursos educativos digitais utilizada para este cenário. Uma rúbrica qualificará se cada uma das etapas do projeto foi cumprida com sucesso.

E como elemento de avaliação sumativa individual, será realizado um quiz em Google Forms, com averiguação dos principais conceitos abordados ao longo deste projeto. O principal critério a aferir será se a capacidade de abstração beneficiou ou não com este cenário de aprendizagem, isto é, verificar se o aluno desenvolveu mapa mental que lhe permita identificar, aplicar, adaptar determinado algoritmo a diversas situações, tendo em atenção que a abstração não é uma *skill* que se dá por adquirida, mas que é um *work-in-progress* constante, sem no entanto esquecer que para se perceber abstração numa determinada linguagem, é de extrema importância uma compreensão sólida dos mecanismos básicos específicos dessa linguagem de programação, nomeadamente, como funções e procedimentos funcionam, particularidade da passagem por parâmetros, por valor e referência, e valores de retorno. Pode-se assim considerar o conhecimento destes conceitos como pré-requisitos, sem os quais não ocorre o desenvolvimento da capacidade de abstração.

Narrativa do Cenário de Aprendizagem

Título: Programar para entreter e abstrair

A narrativa do Cenário deve ser redigida para descrever a visão do ensino-aprendizagem da perspetiva do professor ou da perspetiva dos alunos. Considere-a como uma história que descreve a experiência de aprendizagem. Deve ter cerca de 500 palavras e pode descrever uma experiência de aprendizagem tão longa ou tão curta quanto se pretenda, por vezes numa só aula, mas normalmente abrangendo mais do que uma aula, como por exemplo um projeto cuja conclusão possa demorar várias aulas.

Nunca como nesta década houve uma proliferação de tantas opções de entretenimento como hoje em dia. Somos confrontados com múltiplas formas de distração, que nem sempre são saudáveis, quando expostos e dependentes das mesmas em excesso. Observando o comportamento dos alunos em plena aula, denota-se essa questão desde logo, quer no computador quer de forma mais ubíqua no telemóvel, desta necessidade de estarem constantemente a jogar, como fenómeno de combate ao stress e à ansiedade, como forma de auto-desafio e de recompensa instantânea.

Partindo desta premissa-“problema”, os alunos serão desafiados pelo professor a criarem a sua própria forma de entretenimento jogável, um jogo que replica as funcionalidades do clássico “bangbang”, permitindo a dois jogadores competirem entre si, ao estilo da batalha naval. As variáveis ‘vento’, ‘velocidade’ e ‘direção’ deverão ser tidas em conta. Haverá múltiplas oportunidades de trabalhar ciclos, condicionais, operadores matemáticos e lógicos. Será uma forma também de os alunos perceberem a mecânica base por detrás da criação de um jogo, como a renderização de ecrã e a manutenção de um estado.

O jogo deverá funcionar a partir da consola, permitindo assim também reforçar conteúdos dos módulos anteriores, como leitura e escrita na consola. No entanto, a pedra basilar a ser trabalhada com este cenário será a capacidade de abstração, isto é, a decomposição do todo nas suas partes, bem definidas, organizando melhor o código

escrito. Esta capacidade está identificada como sendo a maior dificuldade dos alunos durante a aprendizagem da programação e é uma das mais importantes para a formação de um profissional na área, e uma das que mais beneficia com um *approach* mais visual, tal como este cenário propõe, e o qual o processo avaliativo aferirá a sua eficácia.

*Este template foi adaptado do modelo de cenário de aprendizagem do **Kit de Ferramentas da Sala de Aula do Futuro**, desenvolvido no âmbito do projeto iTEC (2010-2014) com o apoio do 7.º Programa-Quadro da Comissão Europeia. O kit de ferramentas está disponível em <http://fcl.eun.org/toolkit>*