

3 - TECNOLOGIAS – NA SALA DE AULA E PROJETOS

ABORDAGEM E-SIMPROGRAMMING: PRIMEIRAS ALTERAÇÕES PARA IMPLEMENTAÇÃO EM CONTEXTO ON-LINE E REFLEXÕES.

Daniela Pedrosa - Universidade de Aveiro & CIDTFF; Leonel Morgado - Universidade Aberta; José Cravino - Universidade de Trás-os-Montes e Alto Douro

PALAVRAS CHAVE: SimProgramming; Programação de Computadores; E-learning; Autorregulação e Corregulação das Aprendizagens.

- Contexto em que surge a prática pedagógica / razões para a sua realização;

O insucesso académico em programação de computadores no Ensino Superior é elevado, particularmente na transição da programação inicial para programação avançada. No ensino à distância acresce a necessidade de os alunos terem maiores capacidades de autorregulação e de correção da aprendizagem, por via da autonomia exigida.

Neste âmbito, o projeto SCReLProg (Self and Co-regulation in e-Learning of Computer Programming) visa desenvolver os seguintes objetivos: 1) adaptar ao e-learning o modelo didático SimProgramming (“e-SimProgramming”); 2) identificar e compreender estratégias de autorregulação e correção das aprendizagens de programação em contexto de e-learning..

A primeira interação da investigação decorre na unidade curricular (UC) “Laboratório de Desenvolvimento de Software” (LDS), no 2.º semestre do 2.º ano da Licenciatura de Engenharia Informática da Universidade Aberta, ano letivo 2018/2019, ao longo de 12 semanas letivas, organizadas em 6 tópicos de 2 semanas cada, sobre aspetos de conceção e implementação de software segundo estilos arquitetónicos (MVC, Model-View-Controller).

- Descrição da prática pedagógica (objetivos, público alvo, metodologia, avaliação);

Os objetivos da UC são: 1) contactar com ambientes, ferramentas e métodos de desenvolvimento de software, que permitam a transição de projetos individuais para projetos em equipa, em maior escala; 2) refletir e descobrir as melhores maneiras de desenvolver software, minimizando os custos e maximizando a qualidade do software; 3) aumentar a motivação e o prazer de desenvolver software.

O público-alvo é heterogéneo quanto à idade (entre os 24 anos aos 55 anos); género; residência (diferentes regiões do país e estrangeiro); habilitações académicas (sem licenciatura e licenciados em áreas distintas da informática). Têm em comum serem em geral estudantes já inseridos no mercado de trabalho.

Há dois percursos de avaliação: contínua ou por exame. Na avaliação contínua os alunos puderam optar entre dois percursos de avaliação: a) em equipa (abordagem e-SimProgramming); b) individual (sem essa abordagem). Aos alunos no percurso individual foi dada a opção de participar ou não na investigação, em termos de recolha de dados pessoais, sem, contudo, diferenças pedagógicas decorrentes dessa participação.

A Abordagem e-SimProgramming

A abordagem e-SimProgramming é uma adaptação da abordagem SimProgramming, desenvolvida em contexto presencial (Pedrosa et al., 2016), assente em técnicas de ensino presencial com apoio tecnológico (Kirkwood & Price, 2014). Preconiza atividades de aprendizagem baseada em problemas, em equipa, ao longo de quatro fases (Pedrosa et al., 2017), havendo em cada fase tarefas específicas com durações variadas.

A adaptação ao formato online (“e-SimProgramming”) está em curso. As primeiras alterações decorreram da adaptação ao novo contexto (ensino não presencial assíncrono): diferentes tecnologias de suporte (Moodle); diferente público-alvo (alunos mais velhos que no presencial); diferentes opções didáticas

do docente. Estas últimas centraram-se na metodologia (baseada em projetos em lugar de problemas), no tipo de tarefas (elaboração do projeto e bases teóricas), nos prazos de realização de tarefas, na possibilidade de opção entre dois percursos de avaliação contínua e no peso das atividades de projeto na avaliação final (40%).

- resultados (já recolhidos e esperados);

Nesta primeira tentativa de transposição da abordagem SimProgramming para o contexto online (e-SimProgramming) foram identificados problemas pedagógicos e técnicos:

Problemas pedagógicos:

- 1) Reduzida participação e interação entre alunos (sensação de isolamento);
- 2) Envio tardio das tarefas;
- 3) Não atualização dos perfis de utilizador;
- 4) Falta de aconselhamento proativo individualizado (ex.: alertar os alunos de que não realizaram as tarefas);
- 5) Equívocos de interpretação das tarefas solicitadas aos alunos;
- 6) Inércia na formação de equipas;
- 7) Envio de reflexões quinzenais fora do prazo estipulado e em menor número;
- 8) Baixo grau de eficácia da informação obtida das reflexões quinzenais;
- 9) Baixa perceção da dinâmica de grupo da turma.

Problemas técnicos identificados:

- 1) Inexistência de um sistema de acompanhamento e feedback automatizado para docente e alunos.
- 2) Inexistência de sinalização e alerta nos fóruns que permita ao docente distinguir mais rapidamente entre contributos que carecem de intervenção (e de quem: do docente ou de alunos) e contributos que não carecem;
- 3) Organização visual e de interação da UC que permita melhor perceção da evolução das atividades para docentes e para alunos;
- 4) Pequenos lapsos técnicos diversos.
 - implicações, recomendações, questões em aberto (aplicabilidade a outros domínios científicos, a outros contextos).

Por análise destes problemas, pretendemos reformular o próximo protótipo metodológico da abordagem e-SimProgramming, para melhorar o ensino-aprendizagem, nomeadamente: desenvolvimento de estratégias mais adaptadas ao público-alvo e contexto de ensino; promoção das estratégias de autorregulação e de correção das aprendizagens através da melhoria do design pedagógico; inclusão de ferramentas e soluções tecnológicas.

Pretendemos igualmente analisar o contributo desses processos e ferramentas para o ensino-aprendizagem da programação e o potencial para outras áreas em contexto de e-learning.