

KidCoder: Uma Proposta de Ensino de Programação de forma Lúdica

Cárlisson Borges Tenório Galdino¹, Sebastião Rogério da Silva Neto¹, Evandro de Barros Costa¹

¹Instituto de Computação – Universidade Federal de Alagoas (UFAL)
Caixa Postal 57072-900 – Maceió – AL – Brasil

carlisson@nti.ufal.br, srsn@ic.ufal.br, ebcosta@gmail.com

Abstract. *Computer Science (and similar) courses have presented a troublesome evasion rate. Much of evasion occurs due to difficulties presented by students in disciplines like Logic and Programming. KidCoder is a propose of a serious game in form of a Environment for Practice to motivate students, offering challenges with gradual difficulty, wrapped in an immersive storyline.*

Resumo. *Cursos de Ciência da Computação e afins têm apresentado um índice de evasão preocupante. Grande parte da evasão ocorre devido a dificuldades apresentadas pelos alunos em disciplinas como Lógica e Programação. KidCoder é uma proposta de um serious game na forma de um ambiente de prática, para motivar alunos, apresentando desafios com dificuldade gradual, envoltos em um enredo imersivo.*

1. Introdução

Atualmente a demanda por profissionais de TI é cada vez mais crescente, entretanto, segundo levantamento realizado pela Associação Brasileira de Empresas de Tecnologia da Informação e Comunicação (Brasscom) em 2014 cerca de 78 mil vagas foram oferecidas, destas apenas 33 mil foram preenchidas por profissionais formados em cursos superiores, uma das causas para o não preenchimento dessas vagas é a alta evasão de estudantes dos cursos ligados a computação (BRASSCOM, 2013).

Os fatores que geram os elevados índices de evasão dos cursos da área de computação, referem-se ao fato dos alunos apresentarem dificuldades nas disciplinas que abrangem os conteúdos de raciocínio lógico-abstrato, algoritmos e programação (MCGETTRICK et al. 2004). O trabalho com conteúdo dessa natureza propicia a construção de uma base lógico-matemática, facilitando o processo de aprendizagem de novos conhecimentos em disciplinas que envolvam lógica, inclusive, as de cursos superiores de computação (DIM et al, 2011).

Os professores, no entanto, encontram dificuldades em trabalhar o raciocínio lógico de seus alunos, uma vez que nem sempre dispõem dos recursos didáticos necessários, e os estudantes, por sua vez, nem sempre se sentem motivados a buscar esses conhecimentos. Assim, pesquisadores da área vêm propondo e desenvolvendo, nos últimos anos, ferramentas que estimulem o raciocínio lógico de seus usuários, possibilitando uma aprendizagem mais prazerosa (FRANÇA et al, 2012).

Ao considerar que Programação consiste principalmente em resolver problemas com uso de uma linguagem formal e que solucionar problemas é uma das diversões na classificação de (HUNICKE, 2010) chamada de Challenge, é possível que se encare tal atividade como divertida. Uma vez que se encare a programação como uma tarefa

divertida, pode-se aproveitar de tal abordagem para torná-la mais agradável e envolvente aos alunos.

Desta forma, faz-se relevante a verificação sobre a eficácia do uso de serious game no ensino e na prática de Programação. A proposta deste trabalho é a criação e aplicação de um serious game que se torne um ambiente para a prática de programação. O serious game será um jogo de RPG onde combates e puzzles deverão ser resolvidos através da escrita de código-fonte.

2. Trabalho Relacionados

A fim de contextualizar e fundamentar o presente artigo foram pesquisados trabalhos que foram desenvolvidos ou destinados a facilitar o aprendizado de Programação. Três dos mais importantes são: **Logo**, **Robocode** e **Scratch**, cada um acrescentando seu próprio estilo, com suas qualidades, mesmo que restritas a determinados ambientes.

No artigo de (SILVA, R. M. 2008) é exposto a utilização da linguagem de programação Logo na educação de crianças, bem como as vantagens deste uso para formação do seu intelecto. Torna-se evidente que a linguagem de programação Logo é ótima para o desenvolvimento das habilidades intelectuais da criança, que ao interagir com o computador através da “tartaruga” constrói as estruturas mentais de forma privilegiada, fazendo com que o raciocínio lógico se desenvolva, favorecendo a aprendizagem das matérias curriculares principalmente da matemática.

No trabalho de (O’KELLY et al., 2006) é apresentado a aprendizagem baseada em problemas (PBL). Para funcionar efetivamente a PBL necessita de um ambiente de com um bom conjunto de problemas. Com isso foi utilizado o Robocode como um candidato para PBL. É apresentado a experiência dos alunos ao utilizarem o Robocode e as limitações da mesma.

O trabalho de (SILVA, 2014 *apud* AURELIANO et al, 2012) tem como objetivo avaliar o uso do Scratch como uma abordagem alternativa para o processo de ensino e aprendizagem de programação. Segundo relatos dos autores, o Scratch teve uma influência positiva como ambiente de programação para um curso introdutório em programação e destaca a interface gráfica da ferramenta por permitir que o programa seja visualizado enquanto é executado.

3. Proposta KidCoder

O objetivo deste trabalho é criar um ambiente divertido que estimule a prática de programação, tendo a forma de um serious game mesclando os gêneros de jogos *RPG*, *Puzzle* e *RTS*. (HUNICKE, 2010) elencou oito tipos de diversão que se pode ter. Destes, pretende-se trabalhar prioritariamente:

- **Fantasia** - através da imersão em um mundo fictício, mas coerente e amplo;
- **Narrativa** - o jogo terá uma história maior, com cenas menores enquanto o jogador progride através dele;
- **Desafio** - através de combates e *puzzles*, que exigirão raciocínio do jogador.

Apesar de os trabalhos mais conhecidos que compartilham os mesmos objetivos que este, investirem frequentemente na busca pela diversão através de competição e submissão (dentre os tipos de HUNICKE, 2010), o principal apelo será à imersão. Pretende-se criar um jogo que funcione, como na grande maioria dos *RPGs*,

prioritariamente em modo campanha, apresentando um universo fantástico próprio, com personagens carismáticos e um roteiro de acontecimentos dos quais o jogador deverá participar através de um personagem avatar.

O ambiente para estímulo à programação se dará na solução de conflitos e desafios (*puzzles*): eles serão resolvidos através de programação. Pretende-se que, ao serem apresentados ao jogador fatos e elementos de roteiro instigantes, bem como, desafios cada vez maiores e diversificados, o interesse pelo jogo (em concluí-lo) se mantenha. Como a programação será fundamental para o progresso no jogo, tornando-se gradualmente mais complexa, o jogador deverá evoluir seu raciocínio lógico, evoluindo na proficiência em programação à medida em que seu personagem evolui dentro da trama apresentada. Será adotada programação visual para a construção de algoritmos pelo jogador.

A ambientação do jogo apresenta um universo alternativo. O jogador controla um jovem que, dispondo de um teclado especial capaz de se conectar com diversos dispositivos, deve aprender a programar e se aperfeiçoar, enquanto progride com o objetivo de salvar o mundo da influência de um supercomputador maligno.

Haverá momentos no jogo em que será preciso destravar portas ou organizar dados na memória de robôs para reativá-los, dentre outras atividades, constituindo os *puzzles* do jogo. No mundo de KidCoder existem também robôs de combate controláveis. Na primeira parte do jogo, o personagem terá acesso a robôs com console, de modo que deverá controlá-los em tempo real em combates. Com o adiantar da história, surgirão robôs com slots de memória, permitindo ao jogador escrever rotinas automatizadas de ação, o que habilita, por sua vez, o uso dos robôs contra desafios que exigem mais velocidade de ação e reação.

4. Diferencial da Proposta

A proposta terá como principais influências Pokémon (jogabilidade e estética de movimentação), Logo + jogos de estratégia em tempo real (combates com robôs controláveis) e jogos como Robocode (estética de arena, e combates com robôs programáveis). A intenção é que seu mundo fictício, personagens e enredo despertem interesse e curiosidade no jogador. Os desafios resolvidos via montagem de algoritmo trarão obstáculos, estimulando sua superação. As batalhas se darão com robôs controlados, mas com uso de algoritmos do próprio jogo, permitindo um controle melhor no nível e evolução da dificuldade, sempre adaptada ao momento no jogo, dosada para ser desafiadora sem desmotivar.

4.1. Ambiente de Prática

Quando se fala em jogos voltados para o ensino ou aprendizado, é comum que se pense primeiramente em ambientes de ensino, que transmitam o conhecimento ao jogador. A proposta de KidCoder é oferecer um ambiente de prática. Nesta abordagem, o jogo permite ao usuário desenvolver suas habilidades através de sua interação, melhorando soluções e resolvendo novos problemas.

Há opções de jogos para auxílio no ensino e aprendizado de Programação que oferecem um ambiente de ensino, transmitindo os conceitos todos ao jogador, como uma aula interativa. Os trabalhos relacionados que aqui foram citados oferecem um ambiente de prática, assim como a presente proposta, tendo sido selecionados para ilustrar este artigo por tal semelhança.

4.2. Programação Visual

São duas as possibilidades para um jogo relacionado a programação: a escolha ou criação de uma linguagem; o uso de programação visual. Cada uma dessas abordagens traz suas vantagens e desvantagens.

Programação por código traz, entre suas vantagens, a exigência de uma maior preocupação com detalhes do código como distinção entre maiúsculo e minúsculo, vírgulas ou mesmo a atenção que o usuário deve ter para não trocar letras enquanto escreve um comando. Esta preocupação se torna bastante importante, visto que a atividade de um programador é sempre atenta a tais detalhes.

Por outro lado, o uso de Programação Visual diminui a necessidade de uso do teclado, o que facilitará, futuramente, que o jogo seja portado para equipamentos desprovidos deste acessório, como tablets. Embora a escrita de código representasse uma aproximação do jogo à realidade do programador, os erros de sintaxe podem desmotivar o programador iniciante, caso esta preocupação seja exigida muito precocemente. Ao utilizarmos programação visual, abstraímos os erros sintáticos, permitindo que o jogador foque sua atenção no algoritmo em si, na solução do desafio proposto.

Dada a importância de o jogador ter, ainda assim, algum grau de contato com código-fonte, pretende-se acrescentar, posteriormente, um recurso que apresente ao jogador o código correspondente ao algoritmo visualmente montado por ele.

4.3. Enredo

É comum em jogos que criam ambientes de prática que se apresentem como “brinquedos”, sem apresentar desafios específicos ou um roteiro definido. Esta é uma das diferenças da presente proposta para as soluções aqui apresentadas: KidCoder deverá apresentar um enredo, de modo que haverá desafios específicos a serem superados, encadeados em uma sequência que leve em conta sua dificuldade crescente, ao mesmo tempo alinhados com uma história que envolva o jogador e desperte curiosidade acerca dos próximos acontecimentos. Tal decisão parte da percepção que enredos tendem a aumentar a imersão do jogador, tornando o jogo mais atrativo.

4.4. Desafio Progressivo

Apesar de ambientes de prática permitirem que se construa/proponha desafios dos mais variados, tradicionalmente eles não são apresentados como parte do jogo. É o caso, por exemplo, do Logo. Nada impede que se construa os desafios e os apresente como um documento externo ao jogo (um professor, por exemplo, pode apresentar uma lista de tarefas para uso exclusivo com determinado jogo). Em KidCoder, entretanto, o desafio será progressivo e vinculado ao enredo.

4.3. Conteúdo Programático

Os desafios propostos dentro do jogo estarão conceitualmente associados a tópicos comuns ao Conteúdo Programático de disciplinas de Programação. Em um momento inicial, tal definição ajuda a definir com maior precisão o quanto o jogo pode beneficiar os alunos ao permitir a prática de certo conteúdo.

5. Considerações Finais e Trabalhos Futuros

Pretende-se, com o projeto, oferecer aos alunos de Programação mais uma ferramenta que os motive e lhes permita praticar codificação e montagem de algoritmos. Em um primeiro momento, está sendo construído um protótipo jogável, onde as principais mecânicas de jogabilidade já estejam implementadas.

Há planos de concluir o jogo, incluindo posteriormente um modo tutorial, que facilitaria seu uso por jogadores que nunca tenham tido contato com linguagens ou com o conceito de Programação.

A associação entre os desafios do KidCoder e tópicos comuns a Conteúdos Programáticos de disciplinas de Programação abre espaço para, futuramente, termos um mapeamento do conteúdo abrangido pelo jogo. Futuramente, ao implementar métricas de avaliação de desempenho (a funcionarem de maneira transparente e sutil), será possível ter como resultado uma elaboração de perfil do conhecimento de determinado jogador, sendo capazes de recomendar estudo em áreas de Programação nas quais o jogador mais apresente dificuldade.

Referências

- BRASSCOM - Associação Brasileira de Empresas de Tecnologia da Informação e Comunicação (2013). Disponível em:
<http://www.brasscom.org.br/brasscom/Portugues/>
- SILVA, E. G., Barbosa, A. F., Neto, S. R., HO, R., & Lopes, A. N. Análise de ferramentas para o ensino de Computação na Educação Básica. XXXIV Congresso da Sociedade Brasileira de Computação – CSBC 2014
- DIM, C. A., Rocha, F. E. L. da. APIN: Uma Ferramenta Para Aprendizagem de Lógicas e Estímulo do Raciocínio e da Habilidade de Resolução de Problemas em um Contexto Computacional no Ensino Médio. In: XIX Workshop sobre Educação em Computação, 2011, Natal. Anais do XXI CSBC – 2011.
- HUNICKE, Robin; LEBLANC, Marc; ZUBEK, Robert. Mda: A formal approach to game design and game research. 2010.
- MCGETTRICK, AD, Boyle, R, Ibbett, R, Lloyd, J, Lovegrove, G & Mander, K 2005, 'Grand challenges in computing: Education - a summary' Computer Journal, vol 48, no. 1, pp. 42-48.,10.1093
- O'KELLY, Jackie; GIBSON, J. Paul. RoboCode & problem-based learning: a nonprescriptive approach to teaching programming. **ACM SIGCSE Bulletin**, v. 38, n. 3, p. 217-221, 2006.
- PEREIRA, Pryscilla de Souza; MEDEIROS, Marcos; MENEZES, José Wally Mendonça. Análise do Scratch como ferramenta de auxílio ao ensino de programação de computadores. 2012.
- SOARES DE FRANÇA, Rozelma; COSTA DO AMARAL, Haroldo. Proposta de um Jogo Eletrônico Educativo Aplicado ao Ensino da Lógica de Programação. **Conferencias LACLO**, v. 3, n. 1, 2012.
- SOUZA, Cláudio Morgado de. Visualg – ferramenta de apoio ao ensino de programação. Setembro de 2009.
- SILVA, Ronaldo Machado. O uso da linguagem Logo na educação infantil. Disponível em Acesso em: 23 Mar. 2014.