

Análise Empírica de Jogos Educativos para Dispositivos Móveis voltados a Disseminação do Pensamento Computacional na Educação Básica

Tancicleide C. S. Gomes¹, Andreza L. B. de Alencar¹

¹Departamento de Estatística e Informática – Universidade Federal Rural de Pernambuco – Recife – PE - Brasil

tancigomes@bsi.ufrpe.net, andreza.lba@gmail.com

***Abstract.** The development of computational thinking in basic education is a challenge mainly if we regard the age range of the Early Childhood Education and the Elementary School I. The obstacles range from an appropriate curriculum definition, to the availability of suitable educational resources. In this sense, this paper presents an empirical analyze of three educational games for mobile devices aimed at promoting computational thinking listing its main features, the subject content, user interface elements, and usability limitations.*

***Resumo.** O desenvolvimento do pensamento computacional na Educação Básica é por si só desafiador, sobretudo se considerada a faixa etária abrangida pela Educação Infantil e pelo Ensino Fundamental I. Os obstáculos encontrados englobam desde a definição de um currículo adequado até a disponibilidade de recursos educacionais apropriados. Neste contexto, o presente trabalho analisa de maneira empírica três jogos educacionais para dispositivos móveis voltados a promoção do pensamento computacional, elencando suas principais características, os conteúdos abrangidos, bem como elementos da interface e limitações de usabilidade.*

1. Introdução

A apropriação de jogos digitais em experiências de aprendizagem podem ser substancialmente benéficas por diversos motivos, dentre os quais se pode elencar [Von Wangenheim e Von Wangenheim 2012]: 1. Oferecem um ambiente adequado para que o aluno possa desfrutar de uma ampla experimentação e, assim, visualizar as consequências, aprendendo com seus próprios erros; 2. Permitem que o estudante aprenda fazendo, minimizando o abismo entre a teoria e a prática; 3. Oferecem um *feedback* instantâneo e customizado a cada estudante; 4. Possibilitam com maior facilidade a imersão do estudante no processo de aprendizagem.

Assim, a utilização de jogos digitais tem sido largamente explorada em experiências de ensino do pensamento computacional [de Jesus e Raabe 2010, Dantas, Nogueira e Costa 2012, Sales e Dantas 2010, Souza e Dias 2012]. Compreende-se por pensamento computacional [Von Wangenheim *et al.* 2014]:

[...]uma abordagem para a resolução de problemas de forma que pode ser implementada com um computador, usando um conjunto de conceitos, tais como abstração, recursão e iteração, processamento, análise de dados e criação de artefatos reais e virtuais. [...]

Considerando, por exemplo, iniciativas realizadas na esfera do ensino médio, Gomes e Melo (2013) relatam uma abordagem que envolve tanto atividades sem o uso do computador quanto o uso de jogos digitais. No ensino do Ensino Superior, por exemplo, Rebouças *et al.* (2010) relatam o ensino de programação para estudantes do curso de Licenciatura em Computação através do desenvolvimento de jogos. Por sua vez, de Azevêdo Silva e Dantas (2014), apresentam conteúdos introdutórios de programação utilizando a linguagem *Python* através de um jogo digital intitulado *K-Louro*.

Conforme se pode observar, considerando-se essencialmente o cenário nacional, no âmbito dos relatos apresentados no Simpósio Brasileiro de Informática na Educação (SBIE) ao longo dos últimos cinco anos, predominam iniciativas realizadas no contexto do ensino médio e superior. Há, portanto, pouquíssimos relatos que contemplem iniciativas similares no rol do ensino fundamental e da educação infantil [de Brito Leitão *et al.* 2012, de Almeida, de Lima e Batista 2013].

Desta forma, são necessários artefatos e ferramentas educacionais que suportem o ensino do pensamento computacional desde as séries da educação infantil e ensino fundamental I. E, considerando ainda as premissas da aprendizagem móvel, de modo que o aprendente possa acessar o objeto de estudo a qualquer hora e em qualquer lugar, conforme sua necessidade, emerge a seguinte questão: *Quais jogos educativos para dispositivos móveis seriam adequados para auxiliar no desenvolvimento do pensamento computacional na Educação Infantil e no Ensino Fundamental I?*

Neste sentido, surge o presente trabalho, o qual apresenta uma análise comparativa entre três jogos digitais que se propõem a ensinar conceitos do pensamento computacional. O objetivo consistiu em analisar estes jogos tanto acerca dos conteúdos abrangidos, quanto em relação a maneira como estes conceitos foram apresentados.

Este trabalho está subdividido da seguinte forma: a próxima seção contextualiza esta pesquisa e apresenta os principais trabalhos relacionados à análises empíricas e/ou avaliações de jogos digitais educacionais voltados ao ensino do pensamento computacional. Na Seção 3 são apresentados os métodos e critérios adotados neste estudo. Em seguida, na Seção 4 são apresentados os jogos digitais educacionais utilizados na análise, elencando suas principais características, bem como elementos da interface e limitações de usabilidade. A Seção 5 apresenta os resultados obtidos, já as considerações finais e possíveis trabalhos futuros são apresentados na Seção 6.

2. Currículos para o Ensino do Pensamento Computacional na Educação Infantil e no Ensino Fundamental

O modelo de referência para currículos destinados ao ensino do pensamento computacional CSTA/ACM K-12¹, estabelece objetivos de aprendizagem para a aprendizagem de computação e dentre os tópicos abrangidos está o pensamento computacional.

Neste modelo, o nível 1, que engloba o nível infantil até o sexto ano do ensino fundamental, propõe que os alunos conheçam os conceitos fundamentais de ciência da

¹ Um modelo de currículo proposto fruto de uma parceria entre a *Computer Science Teachers Association* e *Association for Computing Machinery*

computação a partir da integração de competências básicas em tecnologia com ideais simples sobre o pensamento computacional.

A proposta estabelece que as experiências de aprendizagem desenvolvidas, cujo foco deve ser a aprendizagem ativa, criatividade e exploração, devem auxiliar os alunos a enxergar a computação como uma parte importante do mundo em que vivem.

Os objetivos de aprendizagem abrangem tópicos como: *Pensamento computacional, Colaboração, Práticas computacionais e de programação*, dentre outros, os quais envolvem habilidades como:

- ***Pensamento Computacional***
 - [1] *Usar recursos tecnológicos (e.g. quebra-cabeças, programas de pensamento lógico) para resolver problemas apropriados para a idade.*
- ***Práticas Computacionais e de Programação***
 - [9] *Uso apropriado de recursos multimídia (livros interativos e programas educacionais)*
 - [11] *Construir um conjunto de instruções que devem ser executadas para que uma tarefa simples ocorra (e.g. instruir o movimento de uma tartaruga)*
 - [12] *Identificar funções/empregos que usam computação e tecnologia.*

Já o currículo elaborado pela iniciativa *Computing at School*², por sua vez, propõe um conjunto de competências a serem desenvolvidas que são elencadas por conteúdo específico (*algoritmos, dados, computadores*, dentre outros tópicos) e estabelecidas apropriadamente conforme a faixa etária (*key stage*). Dentre as competências estabelecidas pelo *key stage 1*, que abrange a faixa etária dos 5 aos 7 anos, estão: *criar e debugar programas simples, usar o raciocínio lógico para prever o comportamento de programas simples, reconhecer usos comuns de tecnologia da informação além da escola.*

Adicionalmente, são descritos ainda 8 níveis, os quais congregam um conjunto de habilidades esperadas para cada faixa etária. Dos 5 aos 7 anos, estima-se que os alunos avancem do nível 01 ao 03 e ao fim do nível 03, espera-se que eles tenham desenvolvido habilidades como:

- *Controlar brinquedos programáveis para executarem instruções,*
- *Planejar e dar ordens diretas para fazer as coisas acontecerem, como jogar robôs,*
- *Resolverem problemas simples utilizando brinquedos programáveis,*
- *Planejarem uma sequência linear (sem condicionais) de instruções,*
- *Desenvolver e melhorar suas instruções,*
- *Reconhecer que muitos dispositivos do cotidiano respondem a sinais e instruções*

Neste sentido, um conjunto de critérios foi estabelecido a fim de identificar jogos que pudessem auxiliar no ensino do pensamento computacional, considerando o quanto estes jogos possuíam características e elementos que proporcionassem o

² <http://www.computingatschool.org.uk/>

desenvolvimento de algumas das competências e habilidades definidas por estes currículos. Os jogos selecionados, os resultados encontrados e as análises são apresentados nas próximas seções.

3. Busca e Seleção dos Jogos

Os jogos foram buscados diretamente nas lojas de aplicativos *Google Play Store / App Store*, bem como utilizando engenhos de busca na internet (e.g. *Google*) utilizando as seguintes palavras-chave: *programming kids* e *code kids*. Os critérios para a seleção dos jogos digitais a serem avaliados consideraram os seguintes aspectos: a faixa etária abrangida e a quantidade de sistemas operacionais *mobile* para os quais os jogos estivessem disponíveis. No tocante aos sistemas operacionais atendidos, considerou-se que o jogo precisava estar disponível para pelo menos dois sistemas operacionais *mobile*, sendo um deles o *Android*. Este último critério, em especial, deveu-se ao fato de que, no cenário nacional, os *tablets* que vêm sendo distribuídos pelo Ministério da Educação para professores e alunos contém o sistema operacional *Android* [Ghedini 2013].

A faixa etária abrangida foi um dos fatores mais relevantes na escolha, considerando as fases escolares do Infantil II ao 1º ano do Ensino Fundamental I (entre 5 e 7 anos de idade), de modo que quanto menor a faixa etária atendida mais relevante o jogo foi considerado, a partir da premissa que o pensamento computacional deva ser apresentado às crianças tão cedo quanto possível [Gomes e Melo 2013, Silva e Melo 2012, Wing 2006].

A partir destes critérios, foram escolhidos os seguintes jogos para dispositivos móveis: *Lightbot Jr*, *The Foos* e *Robot School*. Embora existam outros aplicativos igualmente relevantes, estes estão disponíveis apenas para o sistema operacional *móvel iOS* (e.g. *iPads* e *iPhones*) e, por este motivo, não foram analisados neste trabalho: *Cargo-Bot*³, *Cato's Hike*⁴, *Code Blast*⁵, *Daisy the Dinosaur*⁶, *Kodable*⁷, *Move The Turtle*⁸.

O idioma não foi considerado um critério relevante pois assume-se que as crianças na faixa etária pretendida não sabem ler (ou não sabem ler com fluidez) e jogos educativos adequados precisam considerar estes aspectos, logo o idioma disponível deva ser praticamente indiferente [Levy 2014].

3.1. *Lightbot Jr*

No *Lightbot Jr*⁹ o jogador controla um robô que necessita transpor obstáculos e iluminar apenas os azulejos azuis. Os comandos são definidos por ícones que ficam dispostos na

³ Site oficial: <http://twolivesleft.com/CargoBot/>

⁴ Disponível na *App Store*: <https://itunes.apple.com/us/app/catos-hike/id574335479?ls=1&mt=8>

⁵ Disponível na *App Store*: <https://itunes.apple.com/br/app/code-blast/id953601961?mt=8>

⁶ Disponível na *App Store*: <https://itunes.apple.com/br/app/daisy-the-dinosaur/id490514278?mt=8>

⁷ Site oficial: <https://www.kodable.com/>

⁸ Site oficial: <http://movetheturtle.com/>

⁹ Site oficial: <http://lightbot.com/>

parte inferior da tela e que são arrastados para o painel de execução (*Main*), um conjunto de instruções a serem seguidos (Figura 01). Os ícones representam ações como: *virar à esquerda*, *virar à direita*, *seguir em frente*, *pular*, *iluminar*, e ainda, outros (não apresentados neste trabalho) que remetem ao uso de procedimentos. À medida que novos comandos estão disponíveis eles são apresentados e explanados ao jogador no início de cada fase (Figura 01). Deste modo, é possível explorar de maneira introdutória conceitos elementares de lógica de programação, *e.g.* reconhecimento de padrões, sequenciamento, *loops*, condicionais, dentre outros.

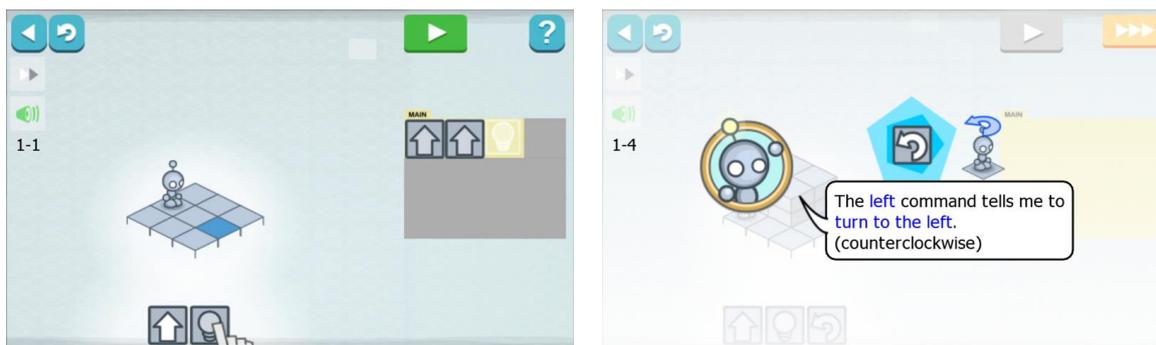


Figura 01. Screenshots do jogo Lighbot Jr

Os elementos da interface são claros para o público alvo em questão, excetuando as setas direcionais que levam algum tempo para serem assimiladas, sobretudo porque as crianças mais novas (4 e 5 anos de idade) ainda estão aprendendo lateralidade, diferentemente das mais velhas com 6 e 7 que já assimilam com maior facilidade. Outro ponto relevante a ser mencionado é que arrastar os ícones até o painel pode ser um fator dificultante, pois o público alvo ainda não tem a coordenação motora bem estabelecida.

Para apoiar os professores, o site oficial disponibiliza alguns recursos complementares como, por exemplo, tutoriais para a iniciação no uso do *Lighbot* pelos alunos em sala de aula. Um outro recurso interessante é que os ícones do jogo são disponibilizados para impressão, o que possibilitaria atividades *semi-desplugadas* envolvendo o *Lighbot*.

3.2. *The Foos*

*The Foos*¹⁰ é voltado para crianças entre 5 e 8 anos. Nele, o jogador deve guiar os personagens para realizar pequenas tarefas ao longo de cada fase, de modo que a dificuldade é aumentada gradualmente e novos recursos e botões de comando são utilizados à medida que o jogador avança (Figura 02).

¹⁰ Site oficial: <http://thefoos.com/>



Figura 02. Screenshot da 1ª fase

Os personagens são controlados pelo jogador através de blocos que representam ações como: *andar*, *pular*, *capturar um objeto*, *lançar um objeto*, e mesmo, instruções algorítmicas mais sofisticadas como: *loops* e *condicionais*, além de comandos extras para personagens específicos.

Os blocos de comando devem ser arrastados até o painel de execução e a execução pode ser acionada tocando no personagem. No momento em que um novo comando é necessário para realizar uma tarefa, uma mão instrui o jogador sobre os novos blocos e/ou parâmetros a serem utilizados, tocando-lhes e arrastando-lhes até o painel de execução. A fase subsequente é sempre similar de modo a reforçar os novos comandos apresentados.

Um ponto considerado negativo, no entanto, são os comandos especiais que, comumente, ficam na lateral esquerda inferior e não são adequadamente apresentados aos jogadores, podendo passar despercebidos e sem nenhuma instrução é possível que os jogadores fiquem sem saber como realizar a tarefa (Figura 03).



Figura 03. Painel de Execução do jogo The Foes

Direcionados aos professores estão disponíveis alguns recursos adicionais no site oficial como o *Teachers Guide*, que contém a resolução dos 24 níveis. Já o *Curriculum*, contém o currículo proposto assim como um breve plano de ensino, demonstrando inclusive recursos exclusivos para professores que estão escondidos no jogo.

3.3. Robot School

No *Robot School*¹¹ o jogador precisa controlar um robô na captura de baterias para recarregar sua energia que é decrescida em uma unidade a cada movimento do robô.

¹¹ Site oficial: <http://www.robotschoolapp.com>

Similarmente aos anteriormente apresentados, os comandos são arrastados para o painel de execução e a medida que os comandos são inseridos o jogador pode visualizar o decréscimo de energia, o que inclusive impede que o jogador insira comandos além da quantidade de energia disponível (Figura 04, Figura 05).

Projetado para crianças acima dos 7 anos de idade, o *Robot School* possui instruções são um pouco mais sofisticadas, tais como o uso de parâmetros. Os comandos novos são apresentados ao jogador no início de cada fase acompanhados de breves instruções, comumente são utilizados duas fases para reforçar o comando novo apresentado. Uma ressalva é que o trajeto é de blocos de montar, o que dificulta a contagem da quantidade de movimentos necessários para chegar em algum ponto da plataforma. Uma possibilidade interessante é a capacidade de interagir com a plataforma em que o robô está, seja girando para os lados, para cima ou para baixo, seja aproximando ou afastando, o que facilita a resolução da tarefa.



Figura 04. Screenshot do jogo Robot School



Figura 05. Screenshots do Painel de Comando e do Código em Swift

Outra possibilidade é visualizar na linguagem *Swift* o código associado a cada bloco, oferecendo uma introdução a programação em linguagens de programação tradicionais menos impactante (Figura 05). Uma limitação, se comparado aos outros

jogos apresentados, é a falta de recursos especialmente destinados aos professores que desejem incorporar o aplicativo em sala de aula.

4. Resultados Obtidos

A partir das habilidades e competências outrora apresentadas, bem como os objetivos de aprendizagem elencados na seção 2, foram estabelecidos alguns conceitos considerados elementares que deveriam ser abrangidos por um jogo educativo que pudesse oferecer suporte a disseminação do pensamento computacional para a faixa etária pretendida.

Os conceitos estabelecidos estão presentes no Quadro 1, com as indicações se o jogo os engloba ou não. Adicionalmente foram incluídos os sistemas operacionais *mobile* para os quais estes jogos estão disponíveis, informações sobre os custos e os idiomas em que estão disponíveis cada um deles (mesmo este aspecto não tendo sido considerado um critério de escolha). Conforme se pode observar os três jogos abrangem alguns dos principais conteúdos atrelados a habilidades consideradas relevantes para a faixa etária atendida, além de estarem disponíveis para os dois principais sistemas operacionais *mobile*.

Quadro 1. Resultados da Análise *Lighbot versus The Foos versus Robot School*

		<i>Lighbot Jr</i>	<i>The Foos</i>	<i>Robot School</i>
CONCEITOS ABRODADOS	Sequenciamento	✓	✓	✓
	Reconhecimento de Padrões	✓	✓	✓
	<i>Loops</i>	✓	✓	✓
	Condicionais	✗	✓	✓
	Funções	✓	✗	✓
	Parâmetros	✗	✓	✗
<i>Sistemas Operacionais Compatíveis</i>				
<i>Gratuidade</i>		✗	✓	✗

Na abordagem proposta pelo jogo *Lighbot Jr*, os níveis são agrupados por conceitos, em que há cinco conjuntos de níveis. Os três primeiros são *Basics*, *Jumping e Overloading*, que abrangem basicamente o sequenciamento de comandos, de modo que os alunos devem executar ações simples, como andar e pular. O *Procedures* apresenta o conceito de funções e de reconhecimento de padrões, de modo que os alunos dispõem de um *slot* de comandos que pode ser retomado ao longo do *slot* principal. Finalmente, o *Loops* retoma todos os conceitos anteriores e apresenta o conceito de repetição que é atrelado ao uso de funções, ao invés de um comando específico de repetição.

The Foos igualmente ao *Lighbot Jr*, inicialmente apresenta o sequenciamento e os alunos podem compreender que necessitam estabelecer um conjunto de passos em uma determinada ordem específica para atingir os objetivos de cada tarefa proposta em

cada nível. O conceito de reconhecimento de padrões e o de repetição surgem a partir do nível 9, em que os alunos precisam compreender que para realizar uma determinada tarefa eles precisam seguir um determinado padrão, que algumas vezes se repete e que pode ensinar a necessidade do bloco de repetição. O bloco de repetição traz arraigado o conceito de parâmetros, onde os alunos podem modificar os ciclos de repetição conforme as opções que variam de 01 a 05 e há também a opção de repetir “infinitamente”. À medida que o jogador avança, algumas fases apresentam mais de um personagem em que o fluxo da realização da tarefa se modifica conforme o personagem, permitindo apresentar o conceito de condicional.

O *Robot School* por ser destinado às crianças a partir dos 7 anos, embora inicie assim como os outros o conceito de sequenciamento já se apropria sutilmente do conceito de parâmetros desde o primeiro nível. Os conceitos de reconhecimento de padrões, *loops* e funções são apresentados conjuntamente em níveis mais avançados, assim como o conceito de condicional.

De um modo geral, um contraponto é o fato de apenas o *The Foos* ser gratuito, o que poderia inviabilizar a utilização dos outros jogos sem apoio financeiro, sobretudo considerando a realidade das escolas públicas.

5. Considerações Finais

A utilização de jogos para auxiliar no desenvolvimento do pensamento computacional na educação básica é crescente ao longo dos últimos anos. No entanto, ainda são *tímidas* as iniciativas voltadas à faixa etária abrangida pela Educação Infantil e o Ensino Fundamental I.

Os jogos apresentados neste trabalho despontam como opções particularmente relevantes para suprir a demanda de artefatos voltados para dispositivos móveis que suportem estratégias e iniciativas para a construção do pensamento computacional. Embora os critérios de escolha referentes aos sistemas operacionais abrangidos tenham limitado substancialmente a diversidade de aplicativos a serem analisados, os jogos apresentados neste trabalho reúnem uma interface gráfica visualmente atraente para o público alvo esperado além de compreenderem um quantitativo significativo de conceitos elementares de lógica de programação.

O escopo deste trabalho restringiu-se a englobar uma análise empírica dos três artefatos, considerando uma avaliação objetiva de caráter qualitativo. Um outro próximo passo relevante consiste em realizar uma avaliação do tipo formativa de cada uma destes artefatos com os usuários finais, a partir de um modelo de avaliação pré-definido, permitindo avaliar a experiência do usuário, a interface, a compreensão dos botões e das instruções fornecidas, dentre outros aspectos.

Referências

- Dantas, V., Nogueira, A. e Costa, T. Gincanas e Jogos Digitais: Usando Competitividade e Tecnologia para Estimular o Raciocínio Lógico. In: Anais do Computer on the Beach, p. p. 131-132, 2012.
- de Almeida M. , L., De Lima C., T. K. e Batista, A. C. D. Pense bem: proposta e desenvolvimento de jogo digital para ensino de computação na educação básica. In: Anais do Simpósio Brasileiro de Informática na Educação. 2013.

- de Azevêdo Silva, M. A., e Dantas, A. (2014). KLouro: Um jogo educacional para motivar alunos iniciantes em programação. In: Anais do Simpósio Brasileiro de Informática na Educação. 2014.
- de Brito Leitão, A. H., Gonçalves, G. G. A., Ribeiro, W. M., de Oliveira, B. C., do Nascimento Salgueiro, V. S. e Madeiro, F. Terra das Cores: Uma Proposta de Jogo Educacional Infantil para o Exercício do Raciocínio Lógico-Matemático. In: Anais do Simpósio Brasileiro de Informática na Educação. 2012.
- de Jesus, E. A. e Raabe, A. L. A. Avaliação Empírica da Utilização de um Jogo para Auxiliar a Aprendizagem de Programação. In: Anais do Simpósio Brasileiro de Informática na Educação. 2010.
- Ghedin, R. (2013) “Estes são os tablets (sim, no plural) da Positivo que o MEC distribuirá aos professores do ensino público”, <http://gizmodo.uol.com.br/tablets-positivo-mec-professores/>. Maio.
- Gomes, T. e Melo, J. O Pensamento Computacional no Ensino Médio: Uma Abordagem Blended-Learning. In: Anais do XXI Workshop sobre Educação em Computação–XXXIII Congresso da Sociedade Brasileira de Computação. Maceió, AL–Brasil. 2013.
- Levay, P. B. (2014) “Utilização de jogos digitais em educação: uma experiência no processo de ensino-aprendizagem de Inglês para crianças”,http://ticeduca2014.ie.ul.pt/downloads/AtasDigitais/Atas_Digitais_ticEDUCA2014.pdf.Maio.
- Rebouças, A. D. D. S., Marques, D. L., Costa, L. F. S. e de Azevedo Silva, M. A. Aprendendo a ensinar programação combinando jogos e Python. In: Anais do Simpósio Brasileiro de Informática na Educação.2010.
- Sales, C. G. e Dantas, V. F. ProGame: um jogo para o ensino de algoritmos e programação. In: Anais do Simpósio Brasileiro de Informática na Educação. 2010.
- Silva, L. K. L da, Melo, J. C. B. de. “Ensino de Computação no Ensino Médio com Abordagem Interdisciplinar”. In: Anais da 64ª Reunião Anual da Sociedade Brasileira para o Progresso da Ciência. São Luis, MA – Brasil. 2012.
- Souza, P. R. A. e Dias, L. R. Kodu Game Labs: Estimulando o Raciocínio Lógico através de Jogos. In: Anais do Simpósio Brasileiro de Informática na Educação, 2012.
- Von Wangenheim, C. G. e Von Wangenheim, A. “Ensinando Computação com Jogos”. Bookess.2012.
- Von Wangenheim, C. G., Nunes, V. R., Santos, G. D., Alves, N. C., Coan, E. S. e Mansur, C. (2014) “Resumo de Objetivos de Aprendizagem de Computação no Ensino Fundamental (Currículo de Referência CSTA/ACM K-12)”, <http://bit.ly/1AEXLrc>, Maio.
- Wing, J. “Computational Thinking”. (2006) In: Communications of the ACM. v.49, n.3. p. 33-35.