

# Um jogo com abordagem interdisciplinar para o ensino do pensamento computacional para crianças

Rozelma Soares de França

Universidade Federal Rural de Pernambuco (UFRPE)  
Universidade Federal de Pernambuco (UFPE)  
Recife – PE, Brasil  
rozelma.franca@ufrpe.br

Victor Afonso dos Santos Ferreira

Universidade Federal de Pernambuco (UFPE)  
Recife – PE, Brasil  
vasf@cin.ufpe.br

## ABSTRACT

The teaching of computational thinking in basic education is urgently needed and digital games can be used to support this process. In this light, this work presents a digital game that aims to promote the learning of computational thinking for children, articulated the teaching of science. The proposal is based on learning discovery and the results of an objective analyze of the game suggest that it conforms to usability heuristics, emerging as a feature supporting the learning of computational thinking.

## RESUMO

O ensino do pensamento computacional na educação básica é uma necessidade premente e jogos digitais podem ser usados para apoiar esse processo. Neste âmbito, este trabalho apresenta um jogo digital que visa promover a aprendizagem do pensamento computacional para crianças, de forma articulada ao ensino de Ciências. A proposta está ancorada na aprendizagem por descoberta e os resultados de uma análise objetiva do jogo sugerem que ele está em conformidade com heurísticas de usabilidade, despontando como um recurso de apoio à aprendizagem do pensamento computacional.

## Categories and Subject Descriptors

K.3.2 [Computers and Education]: Computer and Information Science Education – *computer science education, curriculum*.

## General Terms

Design.

## Keywords

Computational thinking, digital game, discovery learning.

## 1. INTRODUÇÃO

Na sociedade contemporânea é requerido dos estudantes desenvolver diversas habilidades, dentre elas o pensamento computacional. Segundo [14], essa habilidade pode ser definida como um processo de resolução de problemas que inclui diversas características, como ordenar e analisar dados de forma lógica, e criar soluções usando uma série de passos ordenados (algoritmo).

O pensamento computacional é primordial para o desenvolvimento de aplicações tecnológicas, podendo ainda ser usado para apoiar a resolução de problemas de diversas áreas. Assim, é enxergado como uma habilidade fundamental a todos, e não apenas aos cientistas da computação [25].

Nesse contexto, é importante diferenciar Informática de Computação, embora em alguns países essas palavras apresentem o mesmo significado. A Informática está associada ao uso de tecnologia na resolução de problemas. No contexto educacional, seu ensino possibilita a alfabetização digital (*IT literacy*), que já não é mais suficiente no século XXI. A Computação, por outro lado, está particularmente, mas não exclusivamente, relacionada ao estudo, projeto e implementação de sistemas computacionais e aos princípios subjacentes a estes projetos [7]. Na escola, deve-se, então, promover a proficiência digital (*IT fluency*) dos educandos, o que inclui o pensamento computacional.

Como forma de demonstrar que o pensamento computacional pode ser aplicado em diferentes áreas, o relatório do *Workshop on The Scope and Nature of Computational Thinking* [19] traz uma série de cenários. Na medicina, por exemplo, técnicas de sequenciamento comumente empregadas na Computação podem ajudar a melhorar as chances de sucesso na combinação de doadores de órgãos. Em pequena escala, a tarefa de prova cruzada de vários doadores e pacientes é considerada um exercício relativamente simples do pensamento computacional.

Há, assim, um conjunto de conceitos que constituem a base da Ciência da Computação, que tem uma ampla aplicação em múltiplas áreas e que deve ser explorado desde a formação básica habilitando futuros sociólogos, físicos, advogados, dentre outros profissionais, a sintetizar e organizar soluções de problemas de suas respectivas áreas de atuação.

Tais princípios podem ser ensinados aos estudantes de diferentes formas, com ou sem o uso de recursos tecnológicos. Nesse contexto, pode-se destacar o trabalho de [3] que propõem um conjunto de atividades lúdicas envolvendo fundamentos da Ciência da Computação as quais foram publicadas no livro *Computer Science Unplugged*. Tais atividades estimulam o pensamento computacional, sem o uso do computador, e têm sido empregadas com êxito em escolas de diferentes países [23, 20, 24]. Uma das vantagens das atividades supracitadas é a possibilidade de sua aplicação em escolas carentes de recursos tecnológicos.

Outra possibilidade de explorar o pensamento computacional na escola é com o suporte de tecnologias digitais, tais como jogos, os quais já atraem o interesse de crianças. Em um contexto mais amplo, o brinquedo, a brincadeira e o brincar assumem papel fundamental no desenvolvimento humano, estando presentes em todas as culturas e civilizações [17]. O jogo e outros aspectos e dimensões a ele associados estão fortemente relacionados aos processos de aprendizado humano [17, 15]. Aspectos lúdicos diversos que por si só são capazes de engajar os jogadores a

superar obstáculos e atingir objetivos, seja por meio de competição, cooperação ou da combinação desses elementos estão associados aos jogos [17, 15]. E, tais elementos, podem ser explorados no projeto de jogos que busquem promover a aprendizagem de conceitos curriculares.

Na literatura há diversos estudos que versam sobre o desenvolvimento de jogos para o ensino de Computação. Outras metodologias de ensino propõem a construção de jogos digitais, pelos educandos, durante a apropriação de saberes relacionados à Computação. Para tanto, se utilizam de ambientes visuais de programação, tais como Scratch [22] e App Inventor [1].

Quanto a evidências sobre o impacto do uso de jogos digitais na aprendizagem, em uma revisão sistemática de literatura, [18] identificaram que tais recursos impactam positivamente quando introduzidos nos processos de ensino e aprendizagem de programação para iniciantes. Dos estudos primários analisados, apenas um relatou que a construção de jogos pelos estudantes usando um ambiente de programação visual não contribuiu com a aprendizagem. No entanto, os demais estudos, que representam mais de 90% dos analisados, reportaram ganhos na construção de conhecimento de computação.

Considerando o exposto, esta investigação tem como objetivo projetar um jogo digital exploratório para o ensino do pensamento computacional para crianças. Diferente da maioria das propostas existentes propõe-se um recurso que trabalha conceitos de Computação aliado a reflexões sobre meio ambiente, numa abordagem interdisciplinar.

Além desta seção introdutória, este artigo está organizado como segue: a seção 2 apresenta trabalhos correlatos, jogos para o ensino do pensamento computacional. Na seção 3 o jogo proposto nesta pesquisa é apresentado, explanando-se sobre a teoria pedagógica que o embasa, seu enredo e sendo demonstrado um de seus desafios. A seção 4 traz a avaliação objetiva do jogo, na qual foi observado se a proposta atende a heurísticas de usabilidade. Por fim, a seção 5 traz as considerações finais a cerca de todo o trabalho e apresenta alguns trabalhos futuros.

## 2. JOGOS DIGITAIS E PENSAMENTO COMPUTACIONAL

Uma gama de habilidades é exigida das novas gerações, dentre elas o pensamento computacional, que talvez seja a mais importante e menos compreendida [4]. Ele é fundamental a todos e baseia-se em fundamentos da Computação, envolvendo a resolução de problemas, a capacidade de projetar sistemas e a compreensão do comportamento humano [25].

No contexto da educação em computação são percebidos diversos fatores que dificultam a aprendizagem de conteúdos fundamentais da área, como a capacidade de abstrair conceitos na resolução de problemas [10]. Nesse âmbito, os jogos têm ganhado espaço como ferramenta de apoio à aprendizagem, sendo um recurso que pode potencializar o desenvolvimento cognitivo de um indivíduo.

Nos últimos anos, educadores e pesquisadores têm manifestado interesse por jogos voltados ao ensino de Ciência da Computação para crianças e adolescentes, sendo possível identificar diversos trabalhos sobre a temática. Como exemplo,

pode-se citar o trabalho de [8] que propõem um jogo, denominado Robotimov, em que o jogador deve fazer ajustes em códigos para que o comportamento do robô chegue em um determinado estado. Apesar de não ter sido apresentado resultado sobre o uso do Robotiv com estudantes, os autores esperam que a partir da interação com o jogo seja construído conhecimento de Computação, em diferentes níveis cognitivos.

Outro trabalho que pode ser citado é o de [14] que propõem um jogo, chamado GrubiBots, voltado para o público infantil. O GrubiBots permite ao jogador construir algoritmos a partir da interligação entre blocos de programação. Estes blocos são formados por instruções e estruturas de programação. Como destaque, pode-se citar a interface gráfica, a qual adapta-se ao dispositivo usado pelo jogador, podendo ampliar as possibilidades de seu uso.

Ainda nesse contexto, [21] desenvolveram um jogo, denominado Castelo dos Enigmas, que tem como proposta incorporar diversão à aprendizagem de novatos em programação. Tal jogo apresenta desafios que exploram conceitos básicos relacionados à sintaxe das principais estruturas de um algoritmo (repetição, condição), palavras-reservadas e escopo.

Já [27], desenvolveram o Takkou, um jogo voltado ao ensino e a prática de algoritmos e programação. A proposta é baseada na teoria da Aprendizagem Significativa de David Asubel. O Takkou tem características de um puzzle e nele o jogador deve resolver problemas, tendo que refletir sobre uma situação e no menor tempo possível desenvolver uma lógica para solucioná-lo.

Outra proposta consiste no Pense Bem [28], um jogo desenvolvido para crianças a partir de 12 anos que tem como proposta o incentivo à lógica computacional realizada na operação de roteamento e bloqueio de redes. O intuito do jogo é que o jogador trace uma estratégia mental que simula o procedimento realizado logicamente pelos computadores conectados à rede durante um fluxo de dados.

Já [12] propõem o Computino, um jogo educativo baseado no Angry Birds desenvolvido para auxiliar o aprendizado de Números Binários. Nele, o jogador dispõe de um estilingue e pedras que devem ser lançadas sobre os números decimais que, ao serem atingidos, geram pontos ao jogador, possibilitando avançar de nível. O número decimal alvo deve ser o número binário equivalente, exibido na parte superior direita da tela, representado por círculos nas cores verde e laranja. Segundo os autores, por ser constituído de desafios curtos, o Computino é adequado para ser utilizado durante a aula, consolidando a aprendizagem do conteúdo já abordado pelo professor.

Analisando-se experiências de ensino de Computação na educação básica, pode-se perceber o uso recorrente do jogo Light Bot [16]. Nele, o jogador move um robô até o ladrilho azul e acende sua lâmpada. Tais ações são realizadas arrastando-se e soltando-se blocos de comando e à medida que vai atingindo os objetivos de cada fase, o jogador constrói conhecimentos básicos de programação.

O Scratch [16] é outro ambiente bastante utilizado por educadores e pesquisadores. Ele desenvolvido pelo *Lifelong Kindergarten Group* (LLK), grupo de pesquisa do *MIT Media Lab* e possibilita ao usuário interagir e criar animações e vídeos pela combinação de blocos de comando que assemelham-se a peças de

quebra-cabeça. De modo similar, há também o App Inventor [1], que possibilita a construção de aplicativos para dispositivos móveis fazendo uso, também, da programação por blocos.

Na maioria dos trabalhos relatados, não é percebida a preocupação com a interdisciplinaridade do pensamento computacional, a qual pode impactar positivamente na aprendizagem de conceitos computacionais. Nesse sentido, este trabalho buscar preencher essa lacuna com a proposição de um jogo digital exploratório, o qual é descrito na seção a seguinte.

### 3. O BUG NA CIDADE

Buscando promover a aprendizagem do pensamento computacional para crianças, o jogo *Bug na Cidade* foi projetado. Nas subseções seguintes, é apresentada a teoria da aprendizagem que o embasa, assim como seu enredo e desafios.

#### 3.1 Os princípios pedagógicos do jogo

O jogo *Bug na Cidade* visa promover a aprendizagem do pensamento computacional, para crianças, de forma articulada ao ensino de Ciências. Para tanto, explora conceitos de lógica de programação enquanto o jogador realiza coleta seletiva de lixo nos desafios propostos, numa abordagem interdisciplinar.

No que se refere ao pensamento computacional, o jogo possibilita a implementação de soluções de problemas usando a linguagem de programação baseada em blocos, podendo ajudar o estudante na compreensão das etapas básicas para construção de algoritmos. Tal proposta está ancorada no CSTA K-12 Computer Science Standards, em seu nível 1 [29].

Dado a importância do ensino do pensamento computacional, enxerga-se nos currículos escolares brasileiros uma oportunidade de trabalhar tal habilidade articulada com outras áreas. Neste trabalho, a proposta integra o ensino de Computação a um dos temas transversais, Meio Ambiente, definido pelos Parâmetros Curriculares Nacionais do Brasil [5]. Tal proposta está em consonância com a literatura da área de educação em Computação, que preconiza a promoção da aprendizagem do pensamento computacional em uma perspectiva interdisciplinar, onde são construídos conhecimentos de Computação aliado a conteúdos curriculares da educação básica [13].

O jogo proposto tem como âncora a aprendizagem por descoberta, baseada no construtivismo, segundo a qual o estudante deve ser um participante ativo no processo de aprendizagem, experimentando, descobrindo, errando e depurando, e assim construindo um conhecimento significativo sobre os conceitos estudados [6].

Tendo em vista que na aprendizagem por descoberta é esperado que o aprendiz descubra os conceitos de forma autônoma por meio dos artefatos a ele providos, podendo levá-lo a interpretações errôneas ou mesmo a lacunas no conhecimento que ele deveria adquirir, o *Bug na Cidade* integra *feedback* e *scaffolding*. Tais técnicas foram incorporadas como forma de prover orientação para que o estudante construa conhecimento significativo sobre os conceitos abordados no jogo, ao mesmo tempo em que exerce um papel autônomo em seu processo de aprendizagem.

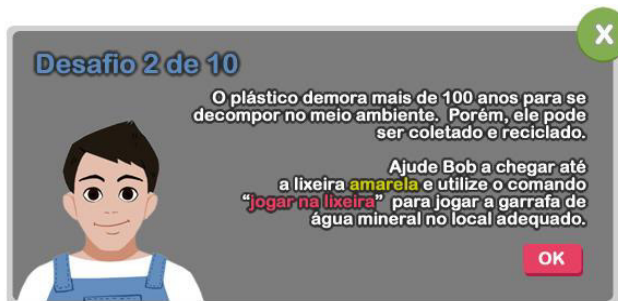
#### 3.2 O nome do jogo

O nome do jogo é um trocadilho: *bug*, termo da língua inglesa, significa ‘besouro’ em português, mas no contexto da Computação significa ‘defeito’, um erro de programação que pode resultar em um comportamento inesperado de um *software*. No contexto do jogo proposto neste trabalho, *bug* é usado para se referir também a um ‘erro’ cometido pelo ser humano, mas com impactos no meio ambiente. O erro é jogar lixo em lugares inadequados e seu impacto é a poluição de uma cidade. Para que tais erros sejam corrigidos, é necessário ao jogador depurá-los e se utilizar da programação de computadores, manuseando blocos de comando, para realizar ações associadas à coleta seletiva de lixo, tornando sua cidade menos poluída e mais sustentável.

#### 3.3 O jogo e seus desafios

O *Bug na Cidade* possui 10 desafios estruturados em níveis de complexidade crescente. Nele, o jogador deve guiar o personagem para realizar a coleta seletiva de lixo. À medida que o usuário avança de fase, novos conceitos computacionais são abordados e informações são introduzidas sobre diferentes tipos de lixo.

(A)



(B)



(C)



Figura 1. A) *Scaffolding*; B) *Feedback de Erro*; e C) *Feedback de Acerto do Desafio 2 do Bug na Cidade*

Os movimentos do personagem são realizados por meio de blocos de comandos que representam ações como: andar, girar para a direita/esquerda, coletar lixo, jogar lixo na lixeira; além de

instruções algorítmicas como estruturas condicionais (se/senão) e laços de repetição (repita).

Os blocos de comando devem ser arrastados até o painel localizado na lateral direita e a execução das instruções é acionada quando o botão verde é clicado. À medida que um novo tipo de lixo domiciliar ou residencial é explorado, informações sobre seu tempo de decomposição no meio ambiente e sugestões de reciclagem, por exemplo, são expostas, como forma de conscientizar o jogador sobre a importância do tema.

Na Figura 1 é ilustrado um dos *scaffolding* associados ao desafio 2 do *Bug na Cidade*. Ele é exibido ao iniciar tal desafio e mantido na barra inferior durante a sua resolução. Além disso, *feedbacks* fornecidos ao jogador após resolver a tarefa e clicar no botão de execução são ilustrados.

## 4. AVALIAÇÃO DO JOGO BUG NA CIDADE

Na literatura há diversas abordagens para avaliação de *softwares* educativos e, em geral, são verificadas características técnicas (usabilidade, qualidade e ergonomia de *software*) e pedagógicas. Como forma de avaliar a qualidade do jogo proposto neste trabalho, uma abordagem objetiva foi adotada, onde especialistas avaliaram os protótipos do jogo por meio de um conjunto de heurísticas de usabilidade de *software*. Esta primeira avaliação objetivou identificar possíveis problemas na interação do usuário com os protótipos até então implementados. A posteriori a proposta será validada com estudantes da educação básica.

### 4.1 Método

A avaliação do *Bug na Cidade* baseou-se na proposta de [26] que propõem um método a partir de análises formativa e objetiva de aspectos pedagógicos e técnicos de um jogo digital para o ensino do pensamento computacional para crianças. No trabalho, os autores partem da premissa de que as heurísticas de Nielsen são formalmente consideradas como um método efetivo de avaliação de interfaces e, tendo em vista as especificidades de *softwares* educacionais, apresentam categorias geradas pela avaliação formativa realizada transpondo as heurísticas de Nielsen para um contexto de *software* educacional.

**Tabela 1. Categorias e Heurísticas para Avaliação de Jogos Exploratórios de Pensamento Computacional [26].**

Categorias	Heurísticas
Interação Exploratória	Ajuda e documentação
	Visibilidade do estado do sistema
	Erros
	Controle e liberdade do usuário
Metáforas Visuais da Interface	Reconhecimento em vez de recordação
Design da Interação	Correspondência entre o sistema e mundo real

As categorias geradas pelo trabalho de [26] são: interação exploratória; metáforas visuais da interface; e *design* de interação. Essas categorias são apresentadas na Tabela 1 e relacionadas com heurísticas de Nielsen conforme proposta dos autores.

## 4.2 Resultados

A partir dos protótipos produzidos, o jogo proposto foi avaliado numa abordagem objetiva. Das impressões registradas pelos pesquisadores durante a análise, nesta seção será apresentada a maneira como o *Bug na Cidade* atende às heurísticas sugeridas por [26].

### 4.2.1 Interação exploratória

#### 4.2.1.1 Ajuda e documentação

Sugere-se que, se necessária, a ajuda seja facilmente encontrada e consultada, sendo objetiva e simples. No *Bug na Cidade* buscou-se projetar interfaces que possibilitem ao jogador interagir com o jogo sem a necessidade de recorrer constantemente à ajuda documentada. Ainda, na ocorrência de dúvidas, o usuário poderá recorrer às instruções sobre como manusear o jogo e como resolver os desafios apresentados. Tais instruções podem ser acessadas ao clique de dois diferentes botões (botão com ícone de interrogação e outro com uma lâmpada, respectivamente), e acionadas a qualquer momento durante a resolução dos desafios.

#### 4.2.1.2 Visibilidade do estado do sistema

Nesse contexto, é recomendado que o sistema mantenha o usuário informado através de *feedback* apropriado, que no contexto educacional pode ser contemplado por processos de *scaffolding*. No *Bug na Cidade* ao iniciá-lo é apresentada ao jogador uma instrução para que ele use os blocos de comando para ajudar o personagem em uma aventura sobre coleta seletiva de lixo. Também, ao iniciar cada desafio seu objetivo é exposto com instruções para que o jogador possa resolvê-lo. Por meio de tais estratégias, espera-se que o usuário explore o jogo de forma mais autônoma, apoiando a aprendizagem por descoberta.

#### 4.2.1.3 Erros

Recomenda-se que o sistema peça confirmações das ações do usuário para prevenir ações indesejadas. No caso de erros, as mensagens devem informar a causa, as consequências e a solução. No contexto educacional, cabe ainda incluir os erros conceituais que podem ser cometidos pelos aprendizes. No caso do *Bug na Cidade*, quando da ocorrência de erro por três vezes consecutivas na resolução de um desafio, é proposto ao jogador um novo desafio em um nível de complexidade inferior.

#### 4.2.1.4 Controle e liberdade do usuário

É sugerido que o sistema proponha meios para o usuário realizar as ações que deseja com possibilidade de desfazê-las e refazê-las. No contexto da aprendizagem por descoberta, essa heurística é de fundamental importância. Nesse sentido, no *Bug na Cidade*, o jogador pode acessar os desafios a qualquer instante, sem haver a necessidade de cumprir um para que outro seja desbloqueado. Ainda, durante a resolução de tais desafios, o jogador poderá fazer e refazer diferentes combinações com os blocos de comando do jogo, interpretando suas consequências.



## 4.2.2 Metáforas visuais da interface

### 4.2.2.1 Reconhecimento em vez de recordação

Nesse contexto é proposto que as informações e objetos relevantes devam estar visíveis para minimizar a carga de trabalho da memória do usuário. Segundo [26] é importante também que os elementos de interface além de identificáveis sejam compreensíveis pelo usuário. No *Bug na Cidade* projetaram-se ícones que remetem às ações de andar para frente, girar para a esquerda/direita, coletar lixo, etc. Tais ícones são padronizados nos desafios e espera-se que os jogadores os reconheçam e use-os adequadamente durante a interação com o jogo, sem a necessidade de redescobrir como usá-los por meio de tentativa e erro.

## 4.2.3 Design da interação

### 4.2.3.1 Correspondência entre o sistema e mundo real

Esta heurística sugere que o sistema fale a linguagem do usuário e use conceitos familiares e convenções do mundo real. Nesse contexto, o *Bug na Cidade* foi projetado contemplando um linguajar mais próximo de seus usuários, crianças. Ainda, buscou-se usar elementos do mundo real na narrativa proposta: o cenário dos desafios é uma cidade e as principais ações do jogador são coletar e jogar o lixo na lixeira, por meio do uso de blocos de comando de programação. Assim, buscou-se propor ações intuitivas, considerando o modelo mental das crianças, no aprendizado de conceitos associados ao pensamento computacional.

## 5. CONSIDERAÇÕES FINAIS

O interesse pelo ensino do pensamento computacional na educação básica é crescente, sendo possível observar iniciativas em diferentes países, incluindo o Brasil. Paralelamente, a literatura da área revela dificuldades na aprendizagem de Computação, ao mesmo tempo em que aponta os jogos digitais como uma ferramenta que pode apoiar os processos de ensino e de aprendizagem de conceitos computacionais.

Considerando o exposto, este trabalho apresentou o *Bug na Cidade*, um jogo que visa promover a aprendizagem do pensamento computacional para crianças de forma articulada ao ensino de Ciências. Tal jogo é baseado na aprendizagem por descoberta e explora conceitos de programação em uma aventura sobre coleta seletiva de lixo. Em uma perspectiva interdisciplinar, apresenta-se, como uma proposta inovadora, que pode promover a aprendizagem de uma habilidade tão requerida na atualidade, o pensamento computacional, associado a reflexões sobre a educação ambiental como tema transversal na escola.

Por meio de uma avaliação objetiva realizada por dois especialistas, autores deste trabalho, pôde-se observar que o *Bug na Cidade* atende a diferentes heurísticas de usabilidade, podendo contribuir para a aprendizagem de conceitos a partir da interação das crianças com o jogo. De modo complementar, pretende-se, como trabalhos futuros, finalizar a prototipação e realizar uma avaliação formativa do jogo proposto. Esta avaliação contará com a participação de estudantes, contemplando aspectos pedagógicos e

técnicos. Assim, será possível, por exemplo, verificar a adequação dos *feedback*, *scaffolding*, ícones, dentre outros, na perspectiva dos usuários. Ainda, será analisado o impacto do *Bug na Cidade* na aprendizagem dos conceitos por ele abordados.

## 6. REFERÊNCIAS

- [1] App Inventor. Disponível em: <<http://appinventor.mit.edu/explore/>>. Acesso em: 24 de agosto de 2016.
- [2] Barbosa, L. S., Fernandes, T. C., & Campos, A. M. (2011). Takkou: uma ferramenta proposta ao ensino de algoritmos. In XVIII Workshop sobre Educação em Computação (WEI 2011).
- [3] Bell, T., Witten, I. H., & Fellows, M. (2011). Computer Science Unplugged. Ensinando Ciência da Computação sem o uso do computador. Tradução por: Luciano Porto Barreto. Disponível em: <<http://csunplugged.org/books>>. Acesso em: 26 de agosto de 2016.
- [4] Blikstein, P. (2008). O pensamento computacional e a reinvenção do computador na educação. Disponível em: <<http://bit.ly/1IXIbNn>>. Acesso em: 26 de agosto de 2016.
- [5] Brasil. Ministério da Educação. Temas Transversais - Meio Ambiente -. Disponível em: <<http://portal.mec.gov.br/seb/arquivos/pdf/meioambiente.pdf>>. Acesso em: 26 de agosto de 2016.
- [6] Bruner, J. S. (1961). The act of discovery. Harvard educational review.
- [7] Computing at School Working Group. (2012). Computer Science: A curriculum for schools. Computing at School Working Group.
- [8] Dantas, V. F., de Macedo, E. R., Andrade, J. R. B., Coutinho, D. R. A., Cavalcante, A. F., Vasconcelos, T. G., & Pereira, M. E. D. S. (2013). Combinando desafios e aventura em um jogo para apoiar a aprendizagem de programação em vários níveis cognitivos. In Anais do Simpósio Brasileiro de Informática na Educação (Vol. 24, No. 1, p. 920).
- [9] de Almeida Melo, L., de Lima Costa, T. K., & Batista, A. C. D. (2013). Pense bem: proposta e desenvolvimento de jogo digital para ensino de computação na educação básica. In Anais do Simpósio Brasileiro de Informática na Educação (Vol. 24, No. 1, p. 346).
- [10] de França, R. S., & Tedesco, P. Explorando o pensamento computacional no ensino médio: do design à avaliação de jogos digitais.
- [11] de Oliveira, G. A., de Bettio, R. W., Rodarte, A. P., & Ferrari, F. B. (2014). GrubiBots Educacional: jogo para o ensino de algoritmos na educação básica. In Anais do Simpósio Brasileiro de Informática na Educação (Vol. 25, No. 1, p. 584).
- [12] França, R. S., Silva, W. C., & Amaral, H. J. C. (2013). Computino: um jogo destinado à aprendizagem de Números Binários para estudantes da educação básica. In Anais do XXI Workshop sobre Educação em Computação-XXXIII Congresso da Sociedade Brasileira de Computação. Maceió, AL-Brasil.

- [13] França, R., & Tedesco, P. (2015, October). Desafios e oportunidades ao ensino do pensamento computacional na educação básica no Brasil. In *Anais dos Workshops do Congresso Brasileiro de Informática na Educação* (Vol. 4, No. 1, p. 1464).
- [14] Google Computational Thinking Repository. Disponível em: <<http://csta.acm.org/Curriculum/sub/CompThinking.html>>. Acesso em: 24 de agosto de 2016.
- [15] Jane, M. (2011). Reality is Broken: Why Games Make Us Better and How They Can Change the World. de Reality is Broken: Why Games Makes Us Better and How They Can Change the World, Penguin, 22-23.
- [16] LightBot. Disponível em: <<http://lightbot.com/>>. Acesso em: 24 de agosto de 2016.
- [17] Ludens, H. (1950). *A Study of the Play Element in Culture*. Boston: Beacon,1.
- [18] Medeiros, T. J., da Silva, T. R., & da Silva Aranha, E. H. (2013). Ensino de programação utilizando jogos digitais: uma revisão sistemática da literatura. *RENOTE*, 11(3).
- [19] National Research Council (US). (2010). *Report of a Workshop on the Scope and Nature of Computational Thinking*. National Academies Press.
- [20] Scaico, P. D., Henrique, M. S., Cunha, F. O. M., & Alencar, Y. M. (2012). Um relato de experiências de estagiários da licenciatura em computação com o ensino de computação para crianças. *RENOTE*, 10(3).
- [21] Scaico, P. D., Marques, D. L., Azevêdo, M. A., Silva, J. C., & Neto, S. V. M. (2011). *Combinando Diversão e Educação: Castelo dos Enigmas, um Jogo Sério para o Ensino de Algoritmos*. XXII SBIE-XVII WIE.
- [22] Scratch. Disponível em: <<https://scratch.mit.edu/>>. Acesso em: 24 de agosto de 2016.
- [23] Sousa, R. V., Barreto, L. P., Andrade, A., & Abdalla, D. (2011). Ensinando e aprendendo conceitos sobre ciência da computação sem o uso do computador: Computação Unplugged!!!. *Jornada de Atualização em Informática na Educação*, 1(1).
- [24] Vieira, A., Passos, O., & Barreto, R. (2013). Um relato de experiência do uso da técnica computação desplugada. In *Anais do XXI Workshop sobre Educação em Computação* (pp. 670-679).
- [25] Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33-35.
- [26] Falcão, Taciana Pontual, and Rafael Barbosa. "" Aperta o Play!" Análise da Interação Exploratória em um Jogo Baseado em Pensamento Computacional." *Anais do Simpósio Brasileiro de Informática na Educação*. Vol. 26. No. 1. 2015.
- [27] Barbosa, L. S., Fernandes, T. C., & Campos, A. M. (2011). Takkou: uma ferramenta proposta ao ensino de algoritmos. In *XVIII Workshop sobre Educação em Computação (WEI 2011)*.
- [28] de Almeida Melo, L., de Lima Costa, T. K., & Batista, A. C. D. (2013). Pense bem: proposta e desenvolvimento de jogo digital para ensino de computação na educação básica. In *Anais do Simpósio Brasileiro de Informática na Educação* (Vol. 24, No. 1, p. 346).
- [29] Seehorn, D., Carey, S., Fuschetto, B., Lee, I., Moix, D., O'Grady-Cunniff, D., ... & Verno, A. (2011). *CSTA K-12 Computer Science Standards: Revised 2011*. Disponível em: <[https://csta.acm.org/Curriculum/sub/CurrFiles/CSTA\\_K-12\\_CSS.pdf](https://csta.acm.org/Curriculum/sub/CurrFiles/CSTA_K-12_CSS.pdf)> Acesso em: 22 de agosto de 2016.