



## Desenvolvimento e Avaliação de um Jogo de Tabuleiro para Ensinar o Conceito de Algoritmos na Educação Básica

### *Development and Evaluation of a Board Game for Teaching the Concept of Algorithms in K-12 Education*

Christiane Gresse von Wangenheim  
Departamento de Informática e  
Estatística – Universidade Federal  
de Santa Catarina (UFSC)  
c.wangenheim@ufsc.br

Giselle Araújo e Silva de Medeiros  
Programa de Pós-Graduação em  
Educação – Universidade Federal  
de Santa Catarina (UFSC)  
giselle.medeiros@posgrad.ufsc.br

Raul Missfeldt Filho  
Departamento de Informática e  
Estatística – Universidade Federal  
de Santa Catarina (UFSC)  
raul.missfeldt.filho@grad.ufsc.br

Giani Petri  
Colégio Politécnico – Universidade  
Federal de Santa Maria (UFSM)  
gpetri@inf.ufsm.br

Fernando da Cruz Pinheiro  
Departamento de Informática e  
Estatística – Universidade Federal  
de Santa Catarina (UFSC)  
fernando.pinheiro@posgrad.ufsc.br

M. Nathalie F. Ferreira  
Programa de Pós-Graduação em  
Design – Universidade Federal de  
Santa Catarina (UFSC)  
nathalie.fortuna@posgrad.ufsc.br

Jean C. R. Hauck  
Departamento de Informática e  
Estatística – Universidade Federal  
de Santa Catarina (UFSC)  
jean.hauck@ufsc.br

### Resumo

Atualmente se observa uma ênfase maior no ensino do pensamento computacional já na educação básica como uma das habilidades do século XXI. O pensamento computacional descreve um modo específico de raciocínio algorítmico baseado em conceitos e processos derivados da programação de computadores. Uma maneira de ensinar conceitos relacionados sobre algoritmos e programação são jogos não-digitais, sendo considerados alternativas de aprendizagem eficazes e ao mesmo tempo divertidas. Nesse contexto, o presente trabalho apresenta *SplashCode*, um jogo de tabuleiro de baixo custo para complementar o ensino de algoritmos e programação, inicialmente apresentado por meio de aula expositiva. O jogo foi desenvolvido de forma sistemática seguindo um processo de design instrucional, e aplicado e avaliado em uma escola pública brasileira. Os resultados da avaliação indicam que o jogo pode contribuir à aprendizagem dos alunos, a interação social, relevância e diversão, proporcionando uma experiência de aprendizagem agradável, lúdica, eficiente e efetiva. O jogo também foi avaliado muito positivamente em relação a usabilidade facilitando o seu entendimento. Um ponto muito elogiado pelos alunos foi o design e a escolha de caracteres *Kawaii* representando adequadamente preferências do público-alvo. Os resultados do presente trabalho podem ser utilizados por professores da educação básica no ensino de computação e por pesquisadores e designers instrucionais para melhorar o desenvolvimento de métodos e materiais instrucionais nesse contexto.

**Palavras-Chave:** Jogo Educacional; Pensamento Computacional; Educação Básica; Programação; Jogo de Tabuleiro

### Abstract

Cite as: Gresse von Wangenheim, C., Medeiros, G. A. e S., Missfeldt Filho, R., Petri G., Pinheiro, F. da C., Ferreira, M. N. F. & Hauck, J. C. (2019). *Development and Evaluation of a Board Game for Teaching the Concept of Algorithms in K-12 Education (Desenvolvimento e Avaliação de um Jogo de Tabuleiro para Ensinar o Conceito de Algoritmos na Educação Básica)*. *Brazilian Journal of Computers in Education (Revista Brasileira de Informática na Educação - RBIE)*, 27(3), 310-335. DOI: 10.5753/RBIE.2019.27.03.310



*Currently, the teaching of computational thinking in K-12 is emphasized as a 21st century skill. Computational thinking describes a specific mode of algorithmic reasoning based on concepts and processes derived from computer programming. One way to teach concepts about algorithms and programming are non-digital games, being considered effective and fun learning alternatives. Within this context, this article presents SplashCode, a low-cost board game to complement the teaching of algorithms and programming initially presented through a lecture. The game was developed in a systematic way following an instructional design process and applied and evaluated in a Brazilian public school. Assessment results indicate that it can contribute to the students' learning, as well as to social interaction, relevance and fun, providing a playful, effective and effective learning experience. The game was also evaluated very positively regarding usability facilitating its understanding. One point highly praised by students was the design and choice of Kawaii characters appropriately representing the target audience's preferences. The results of the present work can be used by K-12 teachers for computing education and by researchers and instructional designers to improve the development of instructional methods and materials in this context.*

**Keywords:** Educational Game; Computational thinking; K-12; Programming; Board Game

## 1 Introdução

Nos últimos anos, há uma crescente ênfase na importância do pensamento computacional como uma habilidade do século 21 (Grover & Pea, 2013; Wing, 2006; Brackmann et al., 2017; Mito et al., 2019). O pensamento computacional denota a ideia de desenvolver uma solução genérica para um problema, decompondo-o, identificando variáveis e padrões relevantes e derivando um procedimento de solução algorítmica (Wing, 2006). Como tal, o pensamento computacional representa uma capacidade cognitiva para aplicar conceitos fundamentais e raciocínio que derivam da computação, em geral, e da programação de computadores, em particular para diferentes outros domínios, incluindo atividades da vida real (Wang, 2015). Assim, o pensamento computacional está presente em processos como pensamento algorítmico, lógica condicional, decomposição, abstração, correspondência de padrões, paralelização, avaliação e generalização (Wing, 2010; Astrachan & Briggs, 2012); refletindo, instanciações cognitivas de conceitos centrais da computação.

Seguindo o CSTA (2017), o pensamento computacional aborda a automação de soluções por meio do pensamento algorítmico, como uma série de passos ordenados, indicando dessa maneira a importância do conceito de algoritmos. Um algoritmo é uma sequência de passos projetados para realizar uma tarefa específica. Os algoritmos são transformados em programas ou códigos-fontes para fornecer instruções para dispositivos de computação. Algoritmos e programação controlam todos os sistemas de computação, permitindo que pessoas se comuniquem com o mundo de novas maneiras e auxiliam na resolução de problemas. Isso demonstra a importância do conceito de algoritmos como uma parte do pensamento computacional (Kalelioglu et al., 2016).

Dessa forma, fomentar o pensamento computacional já desde a educação básica parece desejável, preparando as crianças para as demandas atuais e futuras de nossa sociedade do conhecimento (França & Tedesco, 2015; Brackman et al., 2016). A introdução do pensamento computacional pode proporcionar aos alunos a oportunidade de se envolverem em diversas habilidades do século XXI, incluindo habilidades cognitivas, bem como habilidades e atitudes colaborativas e criativas para resolver problemas (Garneli et al., 2015; Nouri et al., 2019).

Inclusive a Base Nacional Comum Curricular (BNCC) já indica algumas habilidades do pensamento computacional tanto para o Ensino Fundamental como para o Ensino Médio (MEC, 2018) alinhada também as diretrizes da Sociedade Brasileira de Computação (SBC, 2018).

“Associado ao pensamento computacional, cumpre salientar a importância dos algoritmos e de seus fluxogramas, que podem ser objetos de estudo nas aulas de



Matemática. Um algoritmo é uma sequência finita de procedimentos que permite resolver um determinado problema. Assim, o algoritmo é a decomposição de um procedimento complexo em suas partes mais simples, relacionando-as e ordenando-as, e pode ser representado graficamente por um fluxograma.” (MEC, 2018, p. 271)

Atualmente, diversas estratégias instrucionais são adotadas para ensinar o pensamento computacional na educação básica como parte do ensino de computação. E, embora a estratégia mais comum para as atividades voltadas a programação seja usando computadores, muitas vezes são utilizadas também de forma desconectada (sem uso de dispositivos digitais) (Kalelioglu et al., 2016; Tsarava et al., 2017; Brackmann et al., 2017). Essas atividades incluem, por exemplo o uso de jogos de tabuleiro ou cartas, ou atividades cinestésicas, usadas para representar e entender conceitos de pensamento computacional, como algoritmos.

Acredita-se que essas atividades desconectadas, especificamente jogos educacionais, podem trazer diversos benefícios, como o aumento da eficácia da aprendizagem, no interesse e na motivação dos estudantes (Gresse von Wangenheim & Shull, 2009). Os jogos podem criar um ambiente divertido e seguro, em que os alunos podem experimentar alternativas e ver as consequências, aprendendo com seus próprios erros e experiências práticas (Pfahl et al., 2001). A abordagem desconectada também é uma alternativa para ensinar o pensamento computacional em escolas de todo o mundo que não possuem infraestrutura tecnológica básica (Unnikrishnan et al., 2016). Dessa forma, jogos podem ser uma estratégia instrucional eficaz e eficiente para o ensino e aprendizagem.

Nesse contexto, especificamente voltado à aprendizagem de conceitos básicos de algoritmos e programação, já existem diversos jogos de tabuleiro e cartas para crianças a partir de 4 anos. Tipicamente, os jogadores usam cartas de ação (representando movimentos, etc.) para comandar (“programar”) sua ficha sobre um tabuleiro para alcançar uma meta. A maioria está disponível apenas em inglês, dificultando sua aplicação em escolas públicas brasileiras. Além disso, sendo quase exclusivamente desenvolvidos como jogos comerciais, a maioria não apresenta informações sobre como foram desenvolvidos e, mais importante ainda, como foram avaliados. Esse estado da arte atual dificulta a sua aplicação como parte do ensino de computação em escolas brasileiras, principalmente na rede pública. Observando também essa carência de pesquisas científicas nesse contexto, avaliando sistematicamente jogos, as alegações referentes aos seus benefícios são questionáveis (Gresse von Wangenheim & Shull, 2009; Connolly et al., 2012; All et al., 2016). Assim, na prática, o uso de jogos não-digitais para o ensino do pensamento computacional necessita evidenciar a sua qualidade e o impacto esperado na aprendizagem e/ou o engajamento que eles prometem (Caulfield et al., 2011; Connolly et al., 2012; Guzdial, 2008).

Desse modo, para contribuir no ensino do pensamento computacional, e mais especificamente na compreensão de algoritmos, na educação básica no Brasil, é desenvolvido o jogo de tabuleiro gratuito SplashCode. Como parte do seu desenvolvimento, avalia-se também a qualidade desse jogo por meio de um estudo de caso utilizando o modelo de avaliação de jogos educacionais MEEGA+Kids (Gresse von Wangenheim et al., 2018), uma customização do modelo de avaliação de jogos educacionais MEEGA+ (Petri et al., 2018), que tem sido amplamente utilizado na prática (Petri, 2018).

Como resultado dessa pesquisa foi desenvolvido de forma sistemática um jogo educacional, em conformidade com objetivos de aprendizagem propostos pelas diretrizes de currículo da Educação Básica, adotando uma mecânica e narrativa de jogo dentro das preferências do público alvo. Todo o material do jogo, inclusive vídeos explicativos, está disponível gratuitamente online ([http://www.computacaonaescola.ufsc.br/?page\\_id=2703](http://www.computacaonaescola.ufsc.br/?page_id=2703)), possibilitando a sua ampla aplicação em escolas brasileiras. Como resultados da pesquisa, foram



também coletados e analisados dados de um estudo de caso aplicando o jogo, com sucesso, em uma escola de Educação Básica.

## 2 Fundamentação Teórica

### 2.1 Ensino de Computação na Educação Básica

O ensino de computação em escolas deve abordar vários princípios práticos que se referem a comportamentos que alunos, computacionalmente fluentes, utilizam para se envolver plenamente com os conceitos básicos da computação. Atualmente, um número crescente de países está desenvolvendo currículos de referência para guiar o ensino de pensamento computacional na Educação Básica, convergindo em termos de conceitos e técnicas fundamentais e sequenciamento a estágios educacionais, como, por exemplo, no *K-12 Computer Science Framework* (CSTA, 2017). O *K-12 Computer Science Framework* organiza o conhecimento de computação em conceitos fundamentais que representam áreas de conteúdo chave em computação e práticas fundamentais, que representam ações que os alunos usam para se envolver com os conceitos de maneiras ricas e significativas, acrescentando conceitos transversais relacionados a computação (Figura 1). Entre os principais conceitos o *K-12 Computer Science Framework* aborda algoritmos e programação.

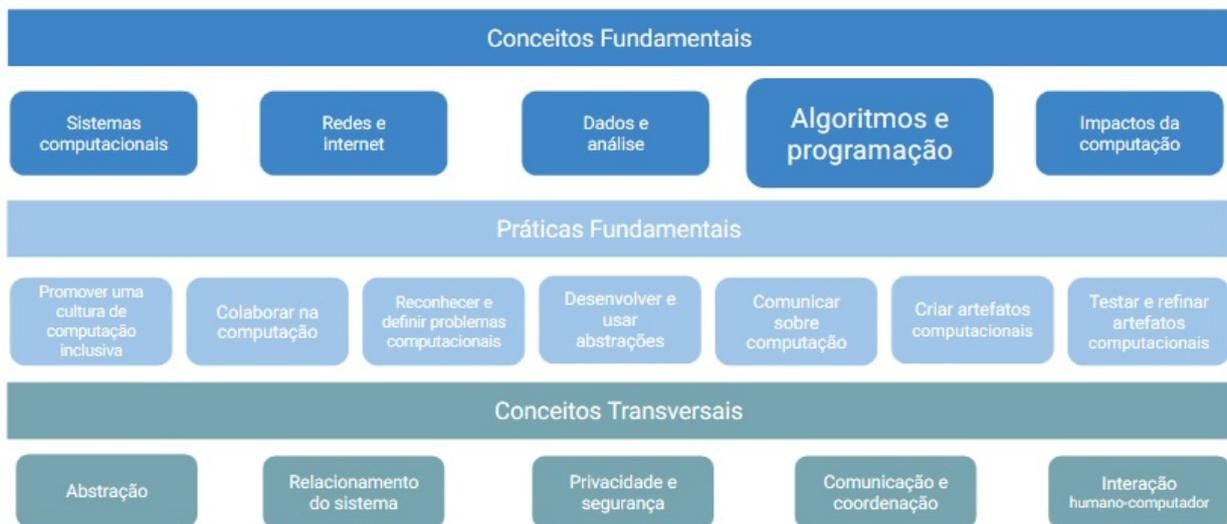


Figura 1: Conceitos e práticas de computação conforme CSTA (2017).

Especificamente em relação a algoritmos, entende-se que são projetados para serem realizados por humanos e computadores. Dessa maneira, alunos nos estágios educacionais iniciais devem aprender sobre os algoritmos apropriados à idade a partir do mundo real. À medida que progridem, os alunos devem aprender sobre o desenvolvimento, combinação e decomposição de algoritmos, bem como a avaliação de algoritmos concorrentes.

No Brasil, a Sociedade Brasileira de Computação (SBC) apresenta diretrizes referentes ao ensino de computação na Educação Básica, que incluem o pensamento computacional (SBC, 2018) além do mundo e a cultura digital (Figura 2). O eixo do pensamento computacional refere-se à capacidade de compreender, definir, modelar, comparar, solucionar, automatizar e analisar problemas de forma sistemática por meio da construção de algoritmos (SBC, 2018). O eixo de cultura digital refere-se ao letramento em tecnologias digitais para comunicação e expressão, e o eixo mundo digital refere-se à apropriação dos processos do mundo digital para compreensão e crítica de tendências (SBC, 2018).

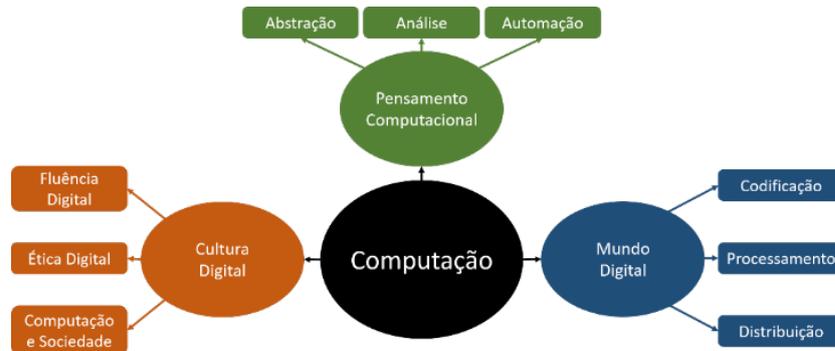


Figura 2: Eixos das Diretrizes da SBC (2018).

Assim, o pensamento computacional envolve abstrações e técnicas necessárias para a descrição e análise de informações e processos e para a automação de soluções. Nesse contexto, considerando um algoritmo como uma descrição de um processo (que resolve um determinado problema). O conceito de algoritmo está presente em todas as áreas e está, intrinsecamente, ligado à resolução de problemas. Seguindo as diretrizes da SBC, espera-se que alunos da Educação Básica aprendam conceitos sobre algoritmos, iniciando com uma definição mais simples no primeiro ano do Ensino Fundamental e aumentando de complexidade a cada ano (Tabela 1).

Tabela 1: Diretrizes da SBC referentes ao objeto de conhecimento de algoritmo no Ensino Fundamental (SBC 2018).

Ano	Objeto de conhecimento	Habilidades
1 Ano	Algoritmo: definição	Compreender a necessidade de algoritmos para resolver problemas. Compreender a definição de algoritmos resolvendo problemas passo-a-passo (exemplos: construção de origamis, orientação espacial, execução de uma receita, etc.).
2 Ano	Algoritmo: construção e simulação	Definir e simular algoritmos (descritos em linguagem natural ou pictográfica) construídos como sequências e repetições simples de um conjunto de instruções básicas (avance, vire à direita, vire à esquerda, etc.). Elaborar e escrever histórias a partir de um conjunto de cenas.
3 Ano	Algoritmo: seleção	Definir e executar algoritmos que incluam sequências, repetições simples (iteração definida) e seleções (descritos em linguagem natural e/ou pictográfica) para realizar uma tarefa, de forma independente e em colaboração.
4 Ano	Algoritmo: repetição	Definir e executar algoritmos que incluem sequências e repetições (iterações definidas e indefinidas, simples e aninhadas) para realizar uma tarefa, de forma independente e em colaboração. Simular, analisar e depurar algoritmos incluindo sequências, seleções e repetições, e também algoritmos utilizando estruturas de dados estáticas.
5 Ano	Algoritmos sobre estruturas dinâmicas	Executar e analisar algoritmos simples usando listas/grafos, de forma independente e em colaboração. Identificar, compreender e comparar diferentes métodos (algoritmos) de busca de dados em listas (sequencial, binária, <i>hashing</i> , etc.).

## 2.2 Jogos Educacionais

Uma abordagem usada para ensinar o pensamento computacional de uma maneira envolvente é a aprendizagem baseada em jogos, que se aproveita da paixão dos alunos pelos videogames (Morrison & Preston, 2009). *Game-based learning*<sup>1</sup> (aprendizagem baseada em jogos) lida com jogos para os quais foram definidos resultados de aprendizagem (Abt, 2002). Esses jogos são projetados para equilibrar o tema didático com a jogabilidade. Um jogo pode ser definido como

<sup>1</sup> Outros tipos de uso de jogos instrucionais, como aprender por meio do desenvolvimento de jogos (por exemplo, ensinar conceitos de computação fazendo com que os alunos programem novos jogos) ou gamificação, estão fora do escopo deste artigo.



qualquer competição entre adversários (jogadores), operando sob restrições definidas (regras), para atingir determinado objetivo (vitória ou recompensa). Nesse sentido, jogos educacionais (ou sérios) são especificamente projetados para ensinar as pessoas sobre um determinado assunto, expandir conceitos, reforçar o desenvolvimento ou auxiliá-las na aprendizagem de uma habilidade ou uma mudança de atitude enquanto jogam.

Jogos podem ser classificados em relação a certas características. O gênero de um jogo é a categorização que agrupa os tipos de jogos de acordo com suas características ou taxonomias (Adams & Rollings, 2006; Djaouti et al., 2011). A plataforma é a classificação dos jogos de acordo com sua estrutura (digitais e não-digitais). Existe uma forte predominância de jogos educacionais digitais disponíveis na literatura, incluindo PC (*stand-alone* e online), *console games* ou *mobile games*. No entanto, jogos não-digitais também são utilizados para fins instrucionais, incluindo jogos de carta, tabuleiro, papel & lápis e acessórios. O modo de interação é uma característica importante dos jogos que permite identificar a interação entre os jogadores. Portanto, o modo de interação é a maneira como o jogador pode interagir com os outro(s) jogador(es) e/ou computador(es)/*console/mobile*.

As regras são restrições contidas no jogo que visam orientar o jogador, determinando condutas corretas ou erradas. As regras são as normas, orientações, fluxo de ações, ou situações que limitam a atuação do jogador dentro do jogo. Elas definem a qualidade do jogo e apresentam um número fixo de diretrizes abstratas que descrevem como o jogo deve funcionar (Salen & Zimmerman, 2009). A mecânica do jogo são as ações do personagem realizadas durante o jogo, elas são regras específicas para limitar o que o personagem pode realizar durante o jogo. A narrativa é uma história com narrações ordenadas, descrevendo acontecimentos ocorrido(s) no(s) cenário(s) onde se passa o jogo. Na narrativa, descreve-se o local onde vivem os personagens, a época e as relações entre os personagens do jogo. A narrativa é o enredo do jogo que precisa motivar o jogador a se tornar cada vez mais imerso (Kremers, 2009).

Os elementos do jogo são os personagens, artefatos ou cenários que integram o ambiente do jogo (Battistella, 2016):

- Personagem: O personagem é um papel fictício utilizando por um ator ao longo do jogo.
- Artefatos: Os artefatos são objetos que terão maior interação com o jogador, normalmente sendo atribuída alguma pontuação.
- Cenários: Os cenários são conjuntos de objetos ou acessórios que ocupam a cena do jogo. Eles são enredos ou ambientes nos quais os jogadores movimentarão seus personagens permitindo, deste modo, prosseguir e avançar no jogo.

### 3 Metodologia

O presente trabalho é classificado como pesquisa exploratória usando uma abordagem de pesquisa multi-método conforme ilustrado na Figura 3.

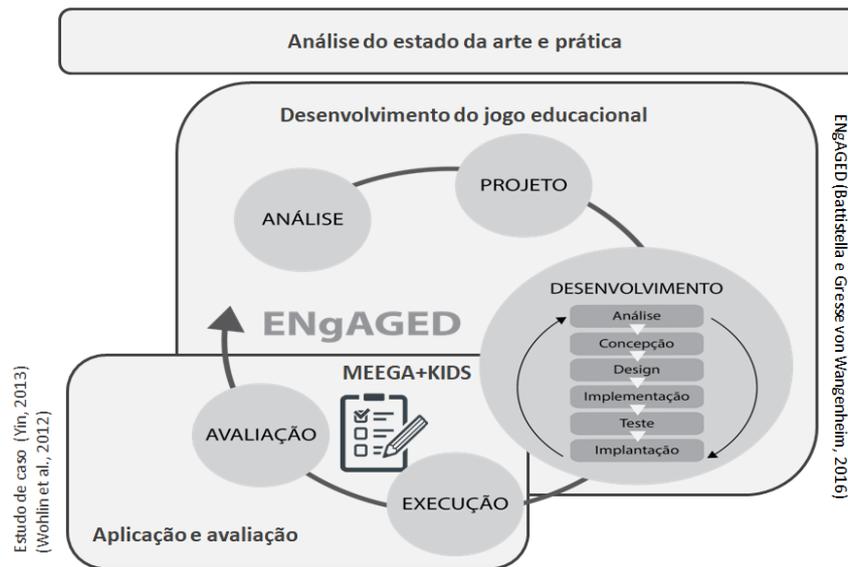


Figura 3: Metodologia de pesquisa.

**Etapa 1. Revisão da literatura.** Inicialmente, analisamos a literatura em relação às atividades educacionais existentes, especificamente jogos não-digitais para o ensino de pensamento computacional. Examinamos artigos publicados em inglês e português disponíveis na Web, por meio das principais bibliotecas digitais e bases de dados da área da informática (IEEE Xplore, ACM, Wiley, Springer e Scopus), com acesso livre por meio do Portal Capes. Para aumentar a cobertura das publicações, também usamos o Google Scholar, que indexa um grande conjunto de dados em várias fontes diferentes (Haddaway et al., 2015). E, como muitos jogos comerciais não são publicados como parte de artigos científicos, também realizamos pesquisas no Google para minimizar os riscos de omissão. Incluímos em nossa revisão apenas jogos não-digitais, de tabuleiro ou carta, com foco no ensino de algoritmos e programação na Educação Básica. Como resultado, encontramos diversos jogos apresentados na Seção 4.

**Etapa 2. Desenvolvimento do jogo educacional.** Seguindo o processo de design de jogos educacionais ENgAGED (Battistella & Gresse von Wangenheim, 2016a) com base no modelo de design instrucional ADDIE (Branch, 2009) o jogo SplashCode foi sistematicamente desenvolvido. Em um primeiro passo foi realizada a definição da meta instrucional, a análise dos aprendizes e do contexto e a definição dos objetivos de desempenho. A partir disso foi projetado o jogo em forma de jogo de tabuleiro e foram desenvolvidos os respectivos materiais do jogo. Durante o desenvolvimento, foram realizados vários testes do jogo para revisar a mecânica e as regras do jogo e para calibrar a duração do jogo. Os resultados gerados durante essa etapa são apresentados na Seção 5.

**Etapa 3. Aplicação e avaliação do jogo educacional.** O jogo foi avaliado por um estudo de caso dentro de um projeto de extensão da iniciativa Computação na Escola/INCoD/INE/UFSC em uma Escola em Florianópolis/SC em 2018-2. O estudo de caso foi sistematicamente definido, planejado, executado e analisado seguindo o processo proposto por Yin (2017) e Wohlin et al. (2012), e avaliado a partir do modelo de avaliação de jogos educacionais MEEGA+KIDS (Gresse von Wangenheim et al., 2018) derivado do modelo MEEGA+ (Petri et al., 2018). O estudo faz parte de um projeto de pesquisa voltado ao ensino de computação na Educação Básica, que foi aprovada pela comissão de ética em pesquisa. Os participantes e seus responsáveis foram esclarecidos em relação ao objetivo da pesquisa e assinaram o termo de consentimento livre e esclarecido/termo de assentimento, autorizando o uso das imagens para fins de publicação científica. Os dados foram coletados usando o



questionário de autoavaliação do MEEGA+KIDS e analisados de forma descritiva. Os resultados da aplicação e avaliação são apresentados na Seção 6.

## 4 Estado da Arte e Prática

Além de um grande número de jogos digitais com foco no ensino de pensamento computacional (Gomes et al., 2016), como por exemplo Lightbot (<https://lightbot.com/flash.html>) ou Programar seu Robô (Kazimoglu et al., 2012), também encontramos vários jogos não-digitais, incluindo jogos de tabuleiro e cartas no nível da Educação Básica, além de diversos jogos educacionais usados no ensino superior (Battistella & Gresse von Wangenheim, 2016b). A Tabela 2 apresenta uma visão geral dos jogos encontrados dentro do nosso foco de pesquisa, incluindo: uma breve descrição, a mecânica e a narrativa do jogo. Identificamos também a idade, idioma e custo buscando a sua aplicabilidade na Educação Básica. Visando a análise da aplicabilidade em aulas no contexto escolar, levantamos também informações referentes ao número de jogadores e à duração da jogada.

Tabela 2: Visão geral sobre jogos de tabuleiro e cartas para o ensino de pensamento computacional na Educação Básica.

Nome	Imagem	Breve descrição	Idade	No. de jogadores	Duração	Idioma	Custo (aprox.)
<b>Bits &amp; Bytes</b> ( <a href="http://www.bitsandbytes.cards">www.bitsandbytes.cards</a> )		O objetivo do jogo é que cada jogador guie seu personagem (programa) para seu planeta natal, Ram, emitindo instruções (vire à direita, vire à esquerda, siga em frente, dê a volta). Ao mesmo tempo, eles precisam evitar paredes, bugs e o temido CPU.	4-9	2-4	--	Inglês	US\$20
<b>C-Jump</b> ( <a href="http://c-jump.com">c-jump.com</a> ) (Singh et al., 2007)		Os esquiadores se alinham no local de partida e correm pelas trilhas de esqui. Espaços no quadro mostram declarações de linguagem de programação. O primeiro jogador a mover todos os esquiadores além da linha de chegada é o vencedor. Os jogadores calculam o número de etapas no movimento, incluindo adição, subtração, divisão e multiplicação de números pequenos. O jogo ajuda a desenvolver a compreensão de um programa de computador completo, formado por sequências lógicas de comandos.	11+	2-4	30 min	Inglês	US\$35
<b>Coder Bunnyz</b> ( <a href="http://www.coderbunnyz.com">http://www.coderbunnyz.com</a> )		Os jogadores programam sua ficha de jogo de coelho robô com seus cartões de código para comer a cenoura e chegar ao destino antes que os outros o façam.	4+	1-4	10-20 min	Inglês	US\$35
<b>Code Island Monkey</b> ( <a href="http://codemoneyplanet.com">http://codemoneyplanet.com</a> )		Como um sábio líder de uma tribo de macacos, os jogadores têm que guiar três macacos em segurança em volta do tabuleiro para o bosque de bananeiras. Portanto, os jogadores têm que usar conceitos como declarações condicionais, loops, booleanos, operadores de atribuição, etc. para mover seus macacos, desviar de armadilhas de areia movediça e marcar frutos ao longo do caminho.	8+	2-4	45 min	Inglês	US\$35



<p><b>Code &amp; Go Mouse Mania</b> (<a href="https://www.learningresources.com/product/code-go-mouse-mania-board-game-2863.do">https://www.learningresources.com/product/code-go-mouse-mania-board-game-2863.do</a>)</p>		<p>Os jogadores desenham cartões de codificação e calculam seus ratos em direção a fatias de queijo espalhadas pelo tabuleiro, navegando em seus camundongos para longe das barreiras e em direção aos túneis de dobra.</p>	5+	2-4	--	Inglês	US\$20
<p><b>Codemaster - programming logic game</b> (<a href="https://www.thinkfun.com/products/codemaster">https://www.thinkfun.com/products/codemaster</a>)</p>		<p>Os jogadores precisam usar a lógica de programação para ajudar seu <i>avatar</i> a coletar os cristais e aterrissar no portal viajando para um mundo exótico. Para todos os 60 níveis, os jogadores terão que usar uma sequência específica de ações para levá-los ao sucesso.</p>	8+	1	--	Inglês	US\$20
<p><b>Coding Farmers</b> (<a href="http://codingfarmers.net">codingfarmers.net</a>)</p>		<p>Os jogadores usam cartões de ação como "avançar por dois espaços" expressos em inglês regular e código Java, rolando um dado e manobrando em torno de obstáculos.</p>	7+	2-4	45-60 min	Inglês	US\$15
<p><b>Crabs&amp; Turtles - Treasure hunt</b> (Tsarava et al., 2018)</p>		<p>Os jogadores têm que manipular peças coloridas representando tartarugas e caranguejos para descobrir a maneira mais eficiente de coletar tesouros colocados nos quadrados da grade do tabuleiro. Para mover um caranguejo ou uma tartaruga, os jogadores precisam criar sequências efetivas de comandos em uma placa de sequência, que representam conceitos de codificação específicos.</p>	6+	equipes de 2	--	Inglês, Alemão, Grego	--
<p><b>Crabs&amp; Turtles - The Race</b> (Tsarava et al., 2018)</p>		<p>Os jogadores têm que chegar ao final do tabuleiro resolvendo e lidando com enigmas e eventos relacionados à matemática. Para fazer isso, os jogadores precisam manipular constantes e alterar valores de variáveis ou tomar decisões baseadas em condicionais.</p>	6+	--	--	Inglês, Alemão, Grego	--
<p><b>FlexiCard</b> (<a href="http://www.computacional.com.br/index.php#FlexiCard">http://www.computacional.com.br/index.php#FlexiCard</a>)</p>		<p>Jogadores se movem num tabuleiro/labirinto para resolver o desafio, como p.ex. chegar até a estrela, pegar a chave e abrir a porta, ou derrubar zumbis e sair pela porta etc. usando as cartas de comando, que são executadas por um jogador mestre.</p>	4+	2-4	15 min	Português do Brasil	Gratuito (CC BY-NS-AS 4.0)
<p><b>Logirunner</b> (Casarotto et al., 2018)</p>		<p>O jogo representa uma corrida em que os jogadores passam instruções de movimento a seus personagens (<i>runners</i>) e estes devem concluir um circuito de objetivos (<i>checkpoints</i>) em um mundo representado por um tabuleiro quadriculado.</p>	6+	2-4	35-65 min	Português do Brasil	--
<p><b>Potato Pirates</b> (<a href="https://www.potatopiratesgame.com">https://www.potatopiratesgame.com</a>)</p>		<p>Os jogadores precisam adquirir todas as cartas rei da batata ou eliminar todos os outros jogadores. Portanto, eles podem assar, esmagar ou fritar seus oponentes e afundar seus barcos usando conceitos de programação como funções, loops e condicionais para fortalecer seus ataques.</p>	6+	3-6	20-40 min	Inglês	US\$30



<b>Robot Turtles</b> <a href="http://www.hinkfun.com/robot-turtles">http://www.hinkfun.com/robot-turtles</a>		<p>Os jogadores ditam os movimentos de suas fichas de tartaruga robô em um tabuleiro de jogo, jogando cartas de código (frente, esquerda e direita). Quando a tartaruga robô de um jogador atinge uma joia, ela ganha. Se cometer um erro, eles podem usar um cartão de <i>bug</i> para desfazer um movimento. O jogo tem muitos níveis, então, conforme os jogadores avançam, eles encontrarão obstáculos como paredes de gelo e usarão códigos mais complexos (como lasers para derreter as paredes).</p>	4+	2-5	15 min	Inglês	US\$25
<b>Robo Rally</b> <a href="https://en.wikipedia.org/wiki/RoboRally">https://en.wikipedia.org/wiki/RoboRally</a>		<p>Os jogadores controlam sua ficha de robô com cartões de jogo que revelam as direções de como os robôs podem se mover pela fábrica de automóveis do <i>Robo Rally</i>. Os jogadores enfrentam obstáculos como lasers industriais, buracos abertos e correias transportadoras em movimento. Para ganhar o jogo, os jogadores precisam ser os primeiros a chegar a cada um dos pontos de controle em ordem numérica.</p>	12+	2-8	45-120 min	Inglês, Francês, Alemão, Holandês	US\$30
<b>Robot Wars</b>		<p>Os jogadores podem atirar nos robôs de outros jogadores durante a guerra. Comandos para <i>bots</i> são escritos em Java, mas legíveis com inglês simples. O primeiro jogador a alcançar a grade central do tabuleiro ganha.</p>	7+	2-4	40 – 60 min	Inglês	US\$20

A maioria desses jogos destina-se a crianças em ensino fundamental (nível da Educação Básica), e são normalmente jogados em pequenos grupos (2-4 jogadores) com duração típica de cerca de uma hora, com poucas exceções de jogos muito rápidos com períodos curtos de cerca 15 minutos. A maioria dos jogos está disponível apenas em inglês, dificultando sua aplicação nas escolas brasileiras. Também observamos que a maioria são jogos comerciais, projetados como jogos familiares. Consequentemente, considerando os preços, a sua aplicação nas escolas pode ser impossível, quando se necessita, por exemplo, 10 jogos para uma turma de 40 alunos. Poucos jogos são desenvolvidos diretamente para um contexto escolar. Observa-se também a escassez de publicações acadêmicas referentes aos jogos, apresentando de forma sistemática como foram desenvolvidos e principalmente avaliados. As poucas avaliações relatadas ((Tsarava et al., 2018; Casarotto et al., 2018; Singh et al., 2007)) foram realizadas com participantes diferentes do público alvo principal, envolvendo somente alunos do ensino superior e não da Educação Básica, deixando questionável a generalização dos resultados obtidos também a alunos mais novos para os quais os jogos são projetados.

## 5 Desenvolvimento do Jogo SplashCode

### 5.1 Análise de Contexto

Seguindo o processo de design de jogos educacionais EnGAGED e o design instrucional são inicialmente estudados os perfis dos aprendizes e o ambiente e assim, a partir desses, definidos os objetivos de aprendizagem.

**Aprendizes:** O público-alvo deste trabalho são alunos da Educação Básica, principalmente do ensino fundamental II (6º a 9º ano), com idade entre 11 e 14 anos. Tipicamente, a maioria dos alunos nessa idade já possuem conhecimentos e habilidades no uso de computadores e outros dispositivos devido ao acesso à internet, redes sociais, jogos digitais,



vídeos e músicas. Os alunos tipicamente também já sabem usar dispositivos digitais (celulares, computadores, *tablets*) em casa e por meio de aulas nas escolas. Gastam um tempo considerável *online* nos seus celulares, principalmente em redes sociais e/ou jogos digitais. No que se refere às competências relacionadas à computação, especificamente em relação à programação, alguns alunos já têm conhecimento sobre o que é um programa de computador, mas poucos sabem programar. Muitos jogam frequentemente videogames, porém poucos têm o costume de jogar jogos não-digitais. Em termos de literatura infantil-juvenil há uma preferência para livros de mangá, anime e a cultura K-pop.

**Ambiente da escola:** O ensino de computação na Educação Básica ocorre atualmente em escolas como parte das aulas ou como atividade extracurricular, tanto por unidades instrucionais explicitamente voltadas ao ensino de computação ou de forma multidisciplinar, integrado em outras disciplinas. Pelo fato de ainda não fazer parte da base curricular no Brasil, essas unidades muitas vezes têm uma curta duração com poucas aulas. As aulas tipicamente ocorrem em salas de informática com computadores (ou *notebooks*). Professores de outras áreas de conhecimento desenvolvem atividades com alunos ensinando o uso de TI (*IT literacy*), como, por exemplo, realizando pesquisas, editando textos, fazendo apresentações, etc. Em algumas instituições há professores de tecnologia educacional na escola, trabalhando com projetos e contribuindo em práticas pedagógicas no uso das mídias em parceria com professores de outras áreas, porém a maioria das escolas no Brasil não possui professores que atuem na área de tecnologia educacional. As turmas possuem em média 25 -40 alunos. Especificamente no contexto de escolas públicas há poucos recursos disponíveis para a aquisição de material didático e/ou material de consumo.

**Currículo:** Segundo o guia de currículo da Sociedade Brasileira de Computação (2018), o ensino de computação na Educação Básica deve abordar como principais eixos o pensamento computacional, mundo digital e cultura digital. Internacionalmente existem diversas iniciativas de definição de currículo voltado ao ensino de computação na Educação Básica. Entre eles, o mais reconhecido é o *K-12 Computer Science Standards* (CSTA, 2016), que apresenta as competências de computação a serem aprendidas no contexto da Educação Básica, incluindo conceitos fundamentais e práticas fundamentais (Tabela 3).

Tabela 3: Competências básicas (CSTA, 2016).

Conceitos Fundamentais	Práticas Fundamentais
Sistema de Computação Redes e Internet Dados e Análise Impactos da Computação <b>Algoritmos e Programação</b>	Promover uma cultura de computação inclusiva Colaborar acerca da computação Reconhecer e definir problemas computacionais Desenvolver e usar abstrações Criar artefatos computacionais Testar e refinar artefatos computacionais Comunicar sobre a computação

Especificamente o conceito de algoritmos e programação deve abordar tópicos incluindo algoritmos, variáveis, controle, modularidade e desenvolvimento de programas. O aluno também deve conhecer o que é e como funciona um algoritmo e princípios da solução de problemas.

O jogo é previsto para ser inserido em uma aula voltada ao ensino de conceitos básicos de algoritmos e programação como método instrucional adicional a aulas expositivas. Essa aula faz parte de atividades de ensino de pensamento computacional, abordando nas demais aulas também o ensino de programação via atividades práticas de programação utilizando ambientes de programação baseados em blocos como Scratch, Snap! e/ou App Inventor.



## 5.2 Design e Desenvolvimento do Jogo SplashCode

De acordo com a análise de contexto é projetado o jogo educacional. O jogo está sendo aplicado em uma unidade instrucional voltada ao ensino de conceitos básicos de computação sobre algoritmos e programação. Esse jogo é desenvolvido com o propósito de completar a explicação sobre conceitos de algoritmos e programação inicialmente apresentados por meio de aula expositiva. Visa-se a adoção de um jogo educacional com o objetivo de proporcionar uma aprendizagem eficaz e ao mesmo tempo divertida, motivando os alunos. De acordo com a disponibilidade de aplicação é definida uma duração máxima para o jogo de 15 minutos. O SplashCode é um jogo de tabuleiro para 2-6 jogadores a partir de 6 anos.

O objetivo do jogo SplashCode é demonstrar conceitos de algoritmos e programação como parte do pensamento computacional. Após a aplicação do jogo SplashCode os alunos da Educação Básica deverão ser capazes de:

- Decompor as etapas necessárias para resolver um problema em uma sequência de instruções.
- Reconhecer que um algoritmo/programa é um conjunto de instruções passo a passo para concluir tarefas.

De acordo com as diretrizes de currículo da SBC (2018) o jogo visa ensinar a compreensão de conceitos básicos referentes a algoritmos:

- Definição: compreender a necessidade de algoritmos para resolver problemas e compreender a definição de algoritmos resolvendo problemas passo-a-passo.
- Construção e simulação: definir e simular algoritmos construídos como sequências e repetições simples de um conjunto de instruções básicas.

Visando um design que seja fácil de aprender a jogar, foi projetado um jogo de corrida (gênero). Optamos pela criação de um jogo não-digital de tabuleiro (plataforma) no modo *multiplayer* para aumentar o engajamento e a interação social dos alunos no início de uma unidade instrucional voltada ao ensino de computação. A escolha de um jogo não-digital também foi motivada pela necessidade de manter o esforço mínimo no desenvolvimento do jogo e reduzir a necessidade de infraestrutura tecnológica para poder jogá-lo.

No contexto da narrativa o jogo, os jogadores precisam levar o seu animal de estimação para casa atravessando uma floresta (o tabuleiro). Eles movem o seu animal por meio de comandos como: mova um para frente, volte, vire à direita, pule etc. Ganha o jogador que primeiro levar o seu animal para casa.

Adotando essa mecânica de jogo, visamos ilustrar aos alunos que algoritmos representam uma sequência de passos, já que os alunos precisam montar a sequência de comandos de movimento para levar o seu animal para casa. Essa mecânica também possibilita aos alunos avaliar se a sua sequência de passos (como as dos outros jogadores) tiveram um resultado de sucesso, por meio do alcance do objetivo de conseguir levar o animal para casa.

Para possibilitar a ampla aplicação em escolas brasileiras, foi projetado um jogo de baixo custo (aprox. R\$10) em língua portuguesa (Tabela 4). De acordo com a preferência observada de mangás nessa faixa etária, foi escolhido um design visual do jogo usando elementos visuais Kawaii. Kawaii é um adjetivo do idioma japonês que pode ser traduzido por “fofo” ou “bonito” e se tornou muito popular entre os adolescentes como uma forma de escrita e



Mascotes “fofos” (Okazaki & Johnson, 2013).

Tabela 4: Material do jogo SplashCode.

Personagens (a serem colados em tampas de garrafa)	
Cartas de movimento	
Quadro de programação	
Tabuleiro	
Vídeo explicativo	

Durante a preparação do jogo, os alunos devem se juntar em grupos de 2-6 jogadores. Cada jogador escolhe um dos animais de estimação para ser seu personagem e recebe um quadro de programação. No início do jogo cada jogador posiciona seu personagem no respectivo campo no tabuleiro. O monte de cartas de programação deve ser embaralhado e colocado ao centro dos participantes.

O jogo é executado em dois passos em cada rodada: programação e execução. No início de cada rodada do jogo, cada jogador recebe 5 cartas de programação. Cada jogador deve então “programar” os próximos 3 passos em seu quadro de programação, colocando em cada espaço uma carta de programação. Concluído o passo da “programação”, começa o passo da execução.



Figura 4: Material do jogo SplashCode.

Começando pelo jogador mais novo, cada jogador deve executar a próxima carta de programação da sua tabela movendo seu personagem pelo tabuleiro de acordo com o comando indicado na carta. Depois ele deve colocar a respectiva carta em um monte de descarte. Caso um jogador colida em um outro personagem, árvore, cair numa poça ou colidir com a lateral do tabuleiro, seu movimento fica bloqueado e a sua personagem permanece no seu campo no tabuleiro. O jogador descarta essa carta de programação e continua no jogo, tentando continuar o seu animal no próximo passo da execução. Esse passo da execução dos comandos “programados” deve se repetir 3 vezes até acabar todas as cartas da tabela de programação dos jogadores. Ao final da rodada, cada jogador deve pegar três novas cartas do monte e programá-las em seu quadro de programação. Esses passos se repetem até terminar o jogo. Ganha o jogo quem primeiro conseguir levar seu personagem ao campo “CASA” do tabuleiro.

Todo o material do jogo SplashCode está publicamente disponível em português no website da iniciativa Computação na Escola ([http://www.computacaonaescola.ufsc.br/?page\\_id=2703](http://www.computacaonaescola.ufsc.br/?page_id=2703)).

## 6 Aplicação e Avaliação

O jogo SplashCode foi aplicado e avaliado no âmbito do projeto Jovens Tutores de Programação com alunos da Escola Básica Municipal Almirante Carvalhal como parte do treinamento dos jovens tutores (10 alunos) e em oficinas de computação com 20 alunos da escola no contraturno escolar durante o semestre 2018-2 (Figura 5).



Figura 5: Alunos jogando SplashCode.



Os participantes foram alunos do ensino fundamental II, com exceção de um aluno de 9 anos (Figura 6a). A maioria dos alunos era de meninas (19). Muitos jogam videogames frequentemente, mas raramente jogos não-digitais (Figura 6b.)

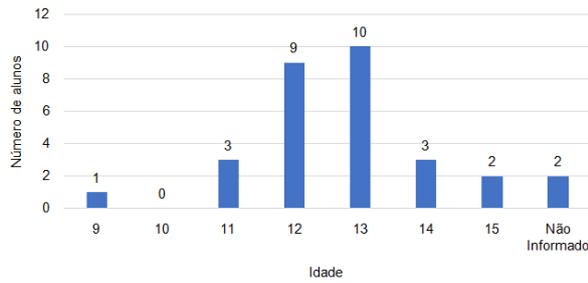


Figura 6a: Idade dos alunos.

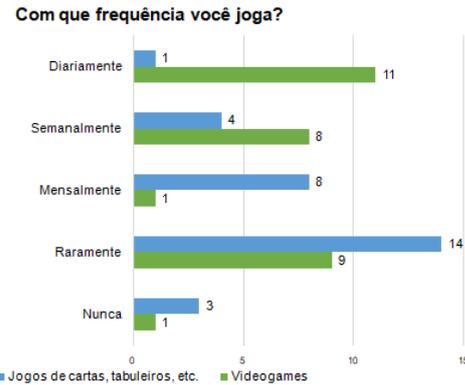


Figura 6b: Hábito de jogar jogos.

## 6.1 Definição da Avaliação

O objetivo da avaliação é analisar se o jogo SplashCode como estratégia instrucional, proporciona uma boa experiência de jogador, usabilidade e possibilita alcançar os objetivos de aprendizagem. Para atingir esse objetivo o jogo foi avaliado por meio de um estudo de caso usando um *research design* não-experimental (apenas uma execução pós-teste), o que permite uma pesquisa aprofundada de um indivíduo, grupo ou evento (Wohlin et al., 2012; Yin, 2017). O estudo é conduzido como um único pós-teste, iniciando com a aplicação do tratamento (jogo educacional SplashCode) e, em seguida, os dados são coletados.

Foi utilizado o modelo de avaliação de jogos educacionais MEEGA+KIDS (Gresse von Wangenheim, et al., 2018), um modelo para a avaliação de jogos educacionais para o ensino de computação no ensino fundamental e médio. MEEGA+KIDS é uma customização do modelo MEEGA+ (Petri et al., 2018), um dos principais modelos de avaliação de jogos educacionais, em termos de usabilidade, experiência do jogador e aprendizagem. Seguindo o modelo MEEGA+, o modelo MEEGA+KIDS é decomposto em fatores de qualidade e suas dimensões (Figura 7). Neste estudo, definimos a usabilidade como grau em que um produto (jogo educacional) pode ser utilizado por usuários específicos (estudantes) para atingir objetivos especificados, com eficácia e eficiência em um determinado contexto de uso (ensino de computação). Assim, a usabilidade é composta das seguintes dimensões: estética, capacidade de aprendizado, operacionalidade e acessibilidade (ISO/IEC, 2014; Davis, 1989; Mohamed & Jaafar, 2010). A experiência do jogador é um fator de qualidade que abrange o envolvimento profundo do aluno na tarefa de jogo, incluindo sua percepção de aprendizagem, sentimentos, prazeres e interações com o jogo, ambiente e outros jogadores (Savi et al., 2011; O'Brien & Toms, 2010; Wiebe et al., 2014; Sweetser & Wyeth, 2005; Fu et al., 2009; Tullis & Albert, 2008; Keller, 1987; ISO/IEC, 2014; Sindre & Moody, 2003). A aprendizagem é medida por meio da correção das respostas de questões múltipla escolha em relação ao objetivo de aprendizagem.

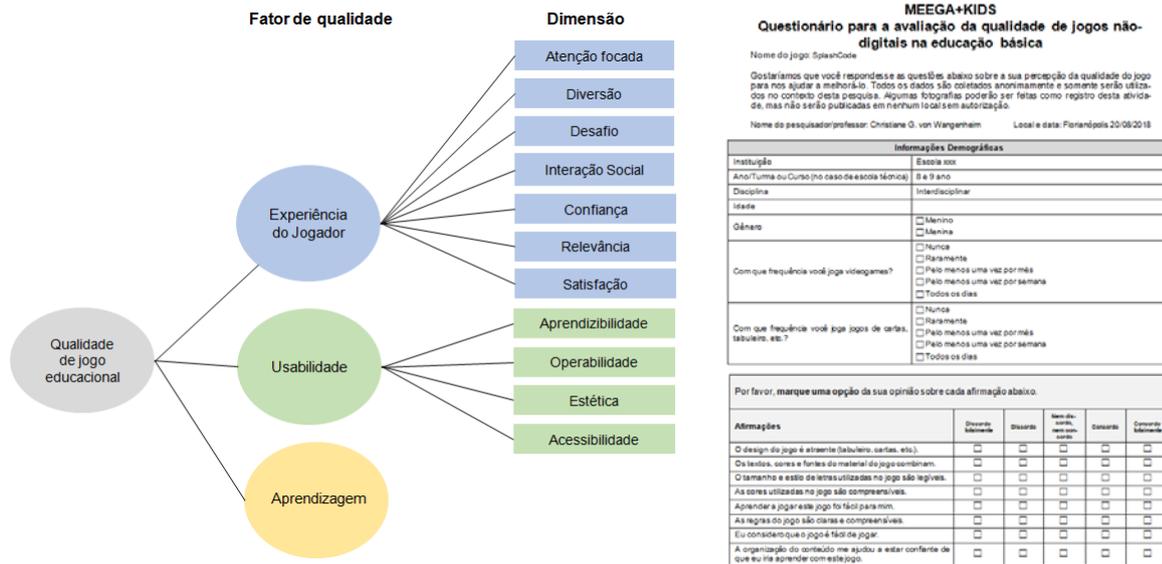


Figura 7: Decomposição do modelo MEEGA+KIDS e extrato do questionário.

A coleta de dados é operacionalizada adotando-se o questionário MEEGA+KIDS (Gresse von Wangenheim et al., 2018) respondido pelos alunos (autoavaliação), a fim de coletar dados sobre suas percepções sobre o jogo, bem como um teste com perguntas de múltipla escolha sobre o assunto. Os dados foram coletados imediatamente após a aplicação do jogo. Foram também coletados comentários qualitativos dos alunos em relação aos principais pontos fortes e fracos do jogo.

## 6.2 Análise dos Dados

Os dados coletados foram transferidos para planilhas e foram gerados gráficos que apresentam a distribuição da frequência e mediana da tendência central das respostas. Os dados coletados no estudo de caso foram agrupados em uma única amostra, utilizando-os cumulativamente para analisar a qualidade do jogo. O agrupamento de dados foi possível devido à similaridade do design de pesquisa das aplicações e a padronização da coleta de dados (Kish, 1994). A este respeito, as aplicações são semelhantes em termos de definição (com o objetivo de avaliar o jogo em relação à experiência do jogador, usabilidade e aprendizagem), design de pesquisa (estudo de caso) e contexto (ensino de computação na Educação Básica). Além disso, houve padronização em termos de medidas (fatores/dimensão de qualidade), método de coleta de dados (questionário do modelo MEEGA+KIDS) e formato de resposta (Escala Likert de 5 pontos) variando de -2 (discordo fortemente) a 2 (concordo fortemente).

### 6.2.1 Avaliação da Experiência do Jogador

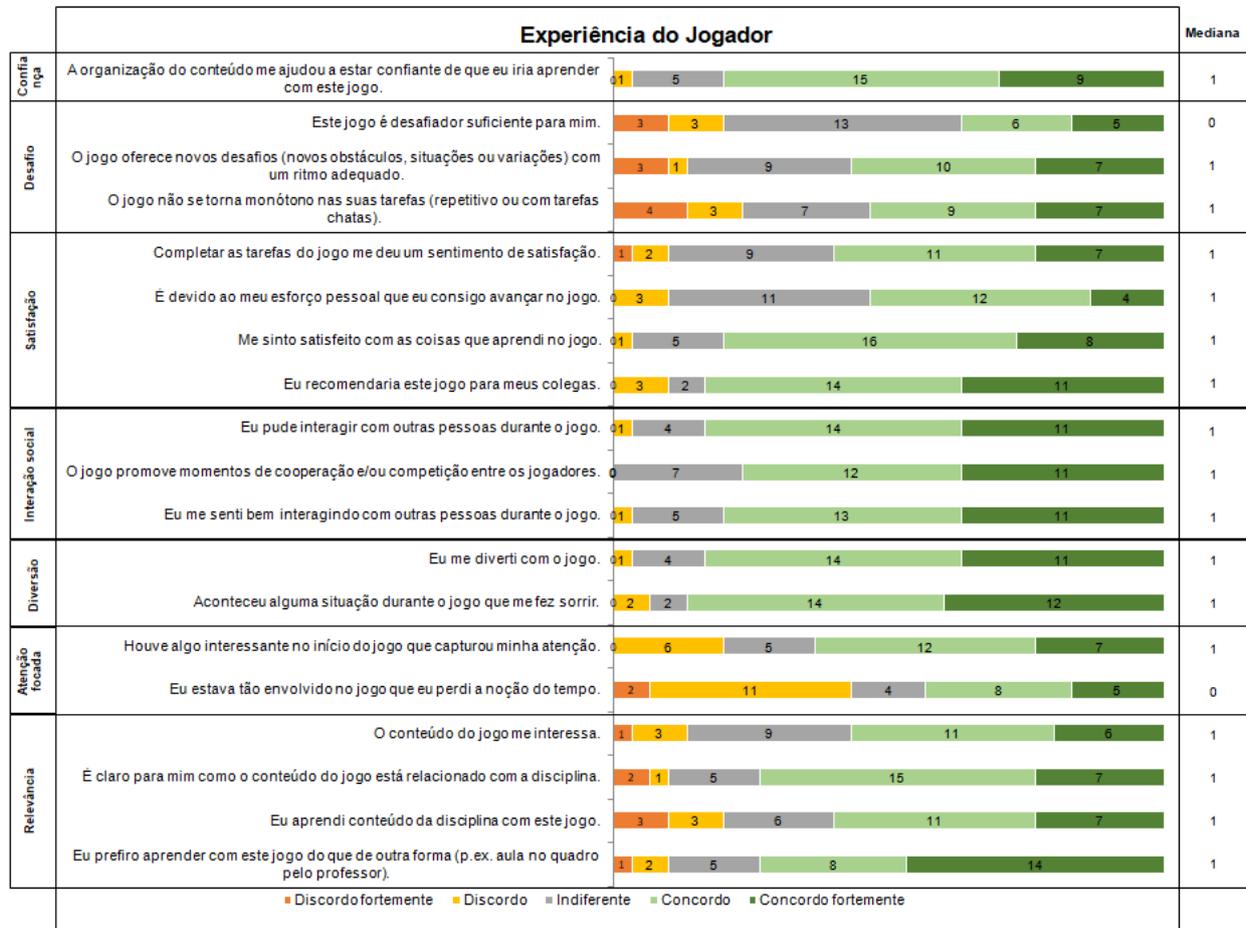


Figura 8: Distribuição de frequência e mediana dos itens referentes à experiência do jogador.

Em geral a experiência do jogador foi avaliada positivamente (Figura 8). As dimensões melhor avaliadas foram a interação social e a diversão. Os alunos também indicaram que o jogo dá a confiança de que estão aprendendo e que o jogo proporciona a satisfação de avançar pela aprendizagem. A grande maioria indicou que recomendaria o jogo aos colegas. Por outro lado, a maioria não se envolveu tanto ao jogo que perdeu a noção de tempo como também indicou que o jogo poderá ser melhorado colocando mais desafios no decorrer da jogada, mesmo sendo um jogo de curta duração. Em termos da avaliação da relevância, também pode ser observada uma quantidade considerável de respostas indiferentes, o que pode ter sido relacionado ao fato que o jogo foi aplicado em oficinas extracurriculares e não no contexto de uma disciplina específica.

### 6.2.2 Avaliação da Usabilidade

Em relação a usabilidade, o jogo foi avaliado de forma muito positiva (Figura 9). A maioria dos alunos achou o jogo fácil de jogar, com regras claras e fáceis de entender. Também concordaram que o design do jogo está atraente e acessível. Sobre o design visual em relação à paleta de cores e tipografia do jogo, foi considerado esteticamente atraente, facilitando sua compreensão. Essa atratividade visual também contribuiu positivamente na facilidade de entender como jogar, além de aumentar a sua aceitação pelos alunos, o que também é evidenciado pelos comentários coletados.

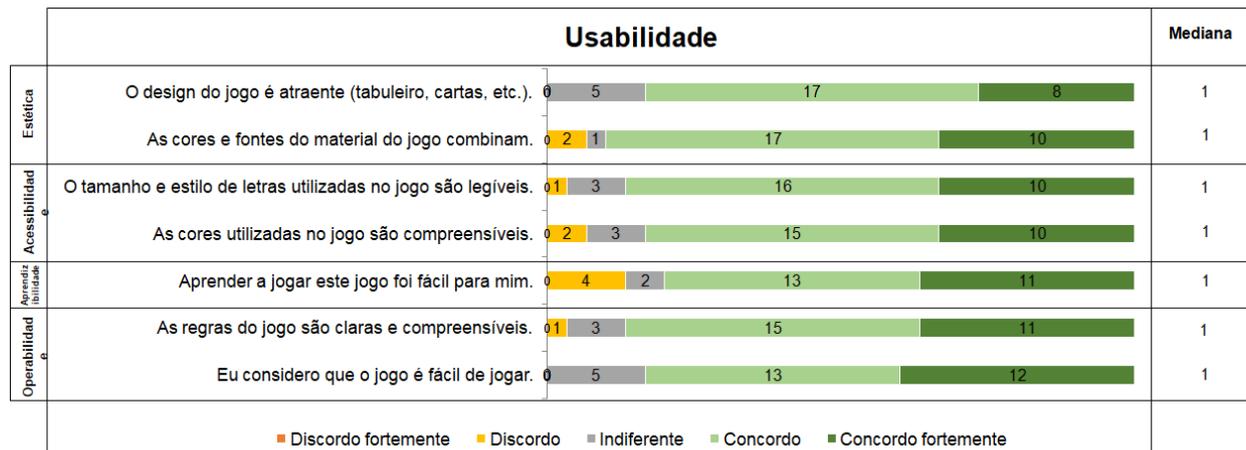


Figura 9: Distribuição de frequência e mediana dos itens referentes à usabilidade.

### 6.2.3 Avaliação da Aprendizagem

Visando a avaliação a aprendizagem, o questionário inclui 3 questões de múltipla escolha referentes ao conhecimento de algoritmos (Figura 10).

**Marque a resposta correta. Cada questão tem uma ÚNICA resposta correta.**

1. Um algoritmo:

- é a parte interna de um computador
- são os passos necessários e ordenados para realizar uma tarefa
- são as imagens que aparecem na tela do monitor
- é um jogo de computador

---

2. Um programa de computador:

- são os dispositivos (como teclado, mouse, etc.) que você usa para interagir com o computador
- são as partes internas do computador (como processador, memória, disco rígido, etc.)
- é a tradução de um algoritmo em instruções que o computador entende
- é o conjunto dos comandos para jogar um jogo no computador

---

3. Considere os seguintes passos de um algoritmo para escovar os dentes:

1. Enxaguar com água.
2. Limpar a escova de dentes.
3. Pegar a escova de dentes.
4. Escovar os dentes.
5. Guardar a escova de dentes.
6. Colocar a pasta de dentes na escova.

Qual a ordem correta dos passos do algoritmo para escovar os dentes?  
(Responda da seguinte forma: por exemplo: 1-2-3-4-5-6).

Resposta: \_ - \_ - \_ - \_ - \_ - \_

Figura 10: Questionário do assunto.

Duas perguntas se referiam a conteúdo diretamente abordado pelo jogo (Questão 1 e 3). A questão 1 foi acertada por quase todos os alunos e a questão 3 pela maioria (60%). A questão 2 que foi acertada por um percentual menor de alunos refere-se a conteúdo mais indiretamente abordado pelo jogo. Esses resultados, mesmo sendo baseado em uma quantidade pequena de questões, e sem pré-teste de comparação, podem fornecer uma primeira indicação de que o jogo ajuda a aprender conceitos relacionados a algoritmos.

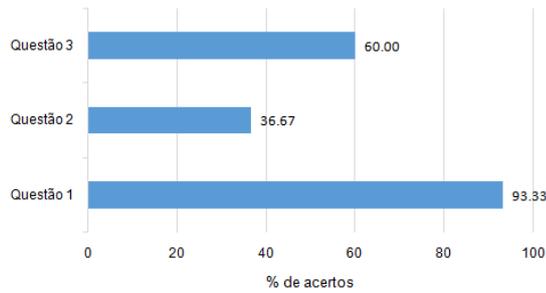


Figura 11a: Porcentagem de acertos para cada questão.

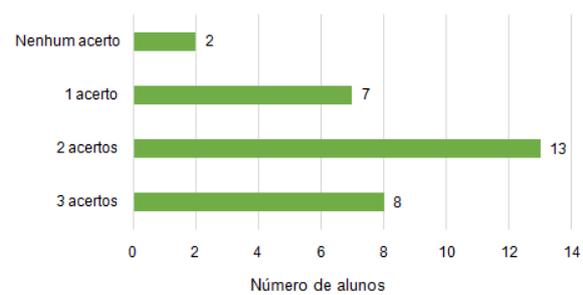


Figura 11b: Quantidade de alunos por quantidade de acertos.

Esses resultados da aprendizagem também são refletidos pelos comentários dos alunos sobre a sua percepção de aprendizagem (Tabela 5). Vários alunos relataram que aprenderam sobre algoritmos, bem como a necessidade de pensar e planejar os passos. Alguns também reconheceram um entendimento sobre como os comandos e, portanto, os programas funcionam. Poucos alunos (3) indicaram que não aprenderam nada jogando o jogo.

Tabela 5: Respostas a pergunta “O que você aprendeu jogando esse jogo?”.

<b>Respostas (indicando as frequências de citação)</b>	
	Pensar no que fazer mais tarde (8)
	Algoritmo (3)
	Como os comandos funcionam (3)
	A programar (3)
	Nada (3)
	Eu aprendi a prestar atenção nos movimentos dos colegas (2)
	Como pode ser divertido jogos de tabuleiro
	Para que possamos ganhar é necessário a colaboração de outros
	Não andamos só para frente
	Não informado (5)

#### 6.2.4 Comentários Gerais

Como principais pontos fortes do jogo, os alunos citaram principalmente a estética dos personagens e o design atraente do jogo. Vários alunos elogiaram, explicitamente, o uso de elementos Kawaii (personagens fofos), indicando assim uma seleção de design de acordo com suas preferências.

A maioria dos estudantes não indicou qualquer necessidade de melhoria. Apenas um estudante considerou o material do jogo ruim. No entanto, vários alunos expressaram uma certa frustração por terem perdido o jogo devido ao azar que tiveram em relação às cartas que receberam aleatoriamente e, assim, limitaram as suas possibilidades de selecionar comandos. Uma solução para reduzir esse impacto da sorte pode ser aumentar o número de cartas que cada jogador recebe no início de cada rodada, aumentando assim as suas alternativas de programação.

Tabela 6: Comentários qualitativos.

<b>O que você gostou no jogo?</b>	Personagens fofos (11) O design (4) Tudo (3) O título do jogo e a ideia do jogo As cores e o visual super kawaii Da jogabilidade Bem divertido Sim Não informado (9)
<b>O que você achou ruim?</b>	Nada (15)



	Ganhar muitas cartas repetidas, fora isso nada Material ruim O jogo foi contra mim Cartas que te permitem ficar preso no início Perder, pois não peguei cartas boas Não informado (10)
<b>Mais algum comentário?</b>	Não (17) Top e fofo Top kawaii Material ruim Não informado (10)

## 7 Discussão

Os resultados indicam uma avaliação positiva do jogo SplashCode. Em particular, os alunos gostaram da interação social e se divertiram enquanto aprendiam com este método instrucional complementar. Observamos também que os alunos participaram ativamente e expressaram ter uma experiência agradável. Isso mostra que o jogo foi percebido mais como uma atividade lúdica e menos como uma atividade instrucional. Esses resultados estão alinhados com o nosso objetivo de ensinar conceitos básicos de pensamento computacional de uma maneira divertida e motivadora. É importante ressaltar que houve a disposição dos alunos em jogar mais vezes o jogo, enquanto esperavam que os outros grupos finalizassem suas partidas, e também a partir da demonstração de interesse em recomendar o jogo para os amigos.

Nesta aplicação inicial, o efeito do jogo referente a aprendizagem dos alunos foi avaliado somente de forma simplificada por meio de poucas perguntas no questionário aplicado após o jogo. Como resultado, observa-se que as duas perguntas (Q1 e Q3) referentes a conceitos abordados no jogo foram respondidas de forma correta pela maioria. Mesmo que o conteúdo da pergunta Q1 sobre o que é um algoritmo tenha sido tratado antes do jogo, a aplicação do jogo pode ter contribuído para ilustrar e fixar esse conteúdo. Especialmente a pergunta Q3, transferindo o conhecimento de sequências de passos aplicados no jogo em relação ao movimento no tabuleiro a uma outra tarefa do dia-a-dia do aluno (escovar dentes), foi respondida corretamente pela grande maioria (mais de 93%) dos alunos. Observamos também que a pergunta Q2, utilizada como controle sobre conteúdo não diretamente abordado pelo jogo, não obteve tantos acertos como as outras perguntas. Esses resultados podem ser uma primeira indicação de que o jogo contribui para a aprendizagem de conceitos de algoritmos. Porém, levando em consideração o caráter inicial desse levantamento do efeito de aprendizagem, necessita-se a replicação da pesquisa com uma coleta de dados mais abrangente e multimodal, para analisar de forma mais detalhada e completa a questão da aprendizagem.

De forma geral, os alunos relataram sentir-se confiantes durante o jogo e perceberam um efeito satisfatório. Embora a maioria dos alunos tenha indicado que sentiram que suas conquistas foram devido ao aprendizado, alguns também comentaram que as chances de vencer o jogo são fortemente influenciadas pelas cartas recebidas. Talvez um agrupamento de tipos de cartas e a distribuição de cartas por grupo possa minimizar esse impacto negativo, por exemplo, impedindo que um jogador receba apenas cartas do mesmo tipo de movimento como 5 cartas "pule obstáculo".

Os alunos também avaliaram a usabilidade do jogo de forma muito positiva. Especialmente o design foi citado várias vezes como ponto positivo do jogo. Vários estudantes também elogiaram a adoção de elementos do Kawaii. Com base em seus comentários favorecendo fortemente os personagens fofos, também inferimos que a narrativa do jogo, ajudando os animais a atravessar a floresta, foi bem aceita. Essa narrativa, com um caráter



positivo de ajudar, foi escolhida intencionalmente, em vez de uma narrativa de combate, como, p.ex., usada no jogo Flexicard. Essa avaliação positiva desse tipo de narrativa e do uso de personagens Kawaii pode ser devida à participação majoritária de meninas em nossas aplicações e pode ser diferente em aplicações com meninos. No entanto, como estatisticamente há um número inferior de meninas na área de computação, essa narrativa e design, aparentemente preferida pelas meninas, pode ser uma estratégia positiva para que meninas passem a ter maior interesse pela área de computação.

Poucos alunos mostraram alguma insatisfação e/ou relataram a percepção de emoções negativas. Apenas um aluno considerou o material como ruim, sem maiores explicações. E, embora, limitando-se a poucas questões de múltipla escolha no final do jogo, identificamos uma primeira indicação de que o jogo pode ajudar a reforçar o entendimento de conceitos de algoritmos no contexto de introdução do ensino de computação.

Durante a primeira aplicação do jogo, percebemos que a explicação inicial de como jogá-lo usando *slides* de forma expositiva, desmotivou parcialmente os alunos. Assim, criamos um vídeo apresentando o jogo de forma lúdica. Na segunda aplicação com o vídeo observamos que isso não apenas reduziu o tempo para a explicação, mas que também deixou os alunos muito mais animados e empolgados em jogar. Foi possível observar que a partir do vídeo, alunos demonstraram compreensão imediata sobre como jogá-lo.

Em geral, a avaliação do jogo produziu resultados promissores em relação à experiência de jogo, usabilidade e aprendizagem, motivando uma aplicação mais ampla deste jogo em atividades de ensino de computação na Educação Básica. Em comparação a jogos semelhantes disponíveis somente em língua estrangeira, o SplashCode se diferencia por estar disponível na Língua Portuguesa para viabilizar a sua aplicação em escolas no Brasil. A principal vantagem do SplashCode em relação a outros jogos disponíveis em português como Logirunner e Flexicards, é a disponibilidade do material na íntegra de forma gratuita no site da iniciativa Computação na Escola/INCoD/INE/UFSC. Destaca-se também a avaliação do jogo em termos da aprendizagem, experiência do jogador e usabilidade, levando em consideração que esse tipo de avaliação sistemática não é apresentado pela maioria dos jogos existentes na forma de soluções comerciais.

**Ameaças à validade.** O design do estudo de caso aplicado nesta pesquisa pode causar vários problemas no que diz respeito à validade dos resultados. Um dos problemas é a falta de base do conhecimento real dos alunos antes de jogar o jogo, para fazer uma comparação e registrar as diferenças. Essa questão também dificulta especialmente a análise de efeitos de aprendizagem obtidos pelo jogo. Assim, os resultados obtidos nessa pesquisa em relação a aprendizagem precisam ser considerados com cuidado, já que representam somente uma primeira indicação. Uma questão a ser analisada também é a confiabilidade e validade dos itens do questionário referentes a aprendizagem, já que esses itens foram especificamente definidos em relação ao objeto de pesquisa (jogo SplashCode), não fazendo parte dos itens padrão do questionário MEEGA+Kids, que foram avaliados em termos de confiabilidade e validade com base no modelo MEEGA+ (Petri, 2018). Porém, como a análise estatística de confiabilidade e validade de instrumentos de coleta de dados necessita um tamanho de amostra considerável, a avaliação da qualidade dessa parte do instrumento está fora do escopo da presente pesquisa exploratória. Prevemos a avaliação dessa questão no longo prazo obtendo uma amostra maior pela replicação do estudo.

Observa-se também, que neste estudo o levantamento de dados referentes a aprendizagem se limitou a poucas perguntas de teste. Para uma reflexão crítica mais completa será importante aplicar também outros instrumentos de pesquisa qualitativa, como grupos focais. Assim, futuros estudos são necessários, incluindo uma coleta de dados mais detalhada e completa, em relação à aprendizagem considerando uma comparação do conhecimento dos alunos antes e depois da



aplicação do jogo.

Isso é ainda agravado pelo fato de não haver um grupo de controle para comparar os efeitos identificados. Neste estudo optamos por realizar um estudo de caso ao invés de um experimento para limitar as interrupções do fluxo normal na unidade instrucional. Por restrições de contexto prático não foi possível a realização de um estudo com maior rigor científico. Assim, os resultados desta pesquisa não podem ser utilizados para comparar os efeitos com outros jogos e/ou atividades instrucionais. Será importante, para analisar mais detalhadamente os efeitos do jogo e compará-los com outros jogos e/ou tipos de atividades instrucionais por meio de novos estudos em forma de experimentos, utilizando no(s) grupo(s) de controle outros jogos e/ou atividades instrucionais.

Além disso, o fato de o jogo ter sido aplicado com um pequeno conjunto de participantes e dentro da mesma escola reduz a possibilidade de generalizar os resultados. No entanto, considerando o caráter exploratório da nossa pesquisa, neste momento, consideramos aceitável o rigor científico de um estudo de caso cuidadosamente definido.

Outra possível ameaça é que aspectos como diversão e satisfação são questões difíceis de medir e capturadas por meio de medidas subjetivas. Para contornar essa ameaça à validade, os itens do questionário foram derivados sistematicamente com base no instrumento de medição padronizado existente no método MEEGA+ que foi avaliado em termos de validade e confiabilidade em larga escala (Petri, 2018).

## 8 Conclusão

Este artigo apresenta o desenvolvimento, aplicação e avaliação de um jogo de tabuleiro de baixo custo voltado a complementar o ensino de conceitos básicos de computação sobre algoritmos. O jogo foi projetado a ser adotado em uma unidade instrucional de forma rápida (aprox. 15 minutos). Resultados de uma aplicação com um total de 30 alunos em uma escola pública em Florianópolis/SC fornecem uma primeira indicação de que o jogo pode contribuir para o aprendizado do entendimento de algoritmos. Os alunos também avaliaram a experiência do jogador e a usabilidade de forma muito positiva, mostrando que o jogo pode ser usado para uma experiência de aprendizagem divertida, eficiente e eficaz.

Com base nesses resultados positivos, estamos disponibilizando todo o material do jogo gratuitamente sob a licença de *Creative Commons* no site da iniciativa Computação na Escola/INCoD/INE/UFSC com o objetivo de viabilizar a adoção do jogo numa escala mais ampla na Educação Básica em escolas brasileiras. Assim, se visa também a ampliação da avaliação da qualidade do jogo, com uma amostra maior e diversificada, em diferentes escolas e localidades no Brasil.

## Agradecimentos

Agradecemos todos os participantes do projeto pelo *feedback*. Este trabalho foi apoiado pelo CNPq, uma entidade do governo brasileiro com foco em desenvolvimento científico e tecnológico.

## Referências

Abt. C.C. (2002). *Serious games*. Lanham: University Press of America. [[GS Search](#)]



Adams, E. & Rollings A. (2006). *Fundamentals of game design*. New York: Prentice Hall. Retrieved from <https://dl.acm.org/citation.cfm?id=1208533>

All, A., Castellar, E. P. N., & Looy, J. V. (2016). Assessing the effectiveness of digital game-based learning: Best practices. *Computers & Education*, 92–93, 90-103. doi: <http://doi.org/10.1016/j.compedu.2015.10.007> [GS Search]

Astrachan, O. & Briggs, A. (2012). The CS Principles Project. *ACM Inroads*, 3(2). [GS Search]

Battistella, P. E. ENgAGED: Um processo de desenvolvimento de jogos para ensino em computação. 2016. Tese (Doutorado em Ciência da Computação), PPGCC/Universidade Federal de Santa Catarina. [GS Search]

Battistella, P. & Gresse von Wangenheim, C. (2016a). ENgAGED: Um Processo de Desenvolvimento de Jogos para Ensinar Computação. In: *Anais do 27º Simpósio Brasileiro de Informática na Educação*, Uberlândia, Brasil. doi:<http://dx.doi.org/10.5753/cbie.sbie.2016.380> [GS Search]

Battistella, P. & Gresse von Wangenheim, C. (2016b). Games for teaching computing in higher education – a systematic review. *IEEE Technology and Engineering Education Journal*, 9(1), 8-30. [GS Search]

Brackmann, C. P., Boucinha, R. M., Román-González, M., Barone, D. & Casali, A. (2017). Pensamento Computacional Desplugado: Ensino e Avaliação na Educação Primária da Espanha. In: *Anais dos Workshops do VI Congresso Brasileiro de Informática na Educação*, Recife, Brasil. <http://dx.doi.org/10.5753/cbie.wcbie.2017.982> [GS Search]

Brackmann, C., Barone, D., Casali, A. Boucinha, R. & Munõz-Hernandez, S. (2016). Computational thinking: Panorama of the Americas. In: *Proc. of the International Symposium on Computers in Education*, Salamanca, Spain. doi: <http://doi.org/10.1109/SIIE.2016.7751839> [GS Search]

Branch, R. M. (2009). *Instructional Design: The ADDIE Approach*. New York: Springer. [GS Search]

Brathwaite, B. & Schreiber, I. (2009). *Challenges for game designers*. Course Technology, CENGAGE Learning, Boston, EUA. [GS Search]

Calderón, A. & Ruiz M. (2015). A systematic literature review on serious games evaluation: An application to software project management. *Computers & Education*, 87, 396-422. [GS Search]

Caulfield, C., Xia, J., Veal, D. & May, S. P. (2011). A systematic survey of games used for software engineering education. *Modern Applied Science*, 5(6), 28-43. [GS Search]

Casarotto, R. I., Bernardi, G., Cordenonsi, A. Z. & Duarte Medina, R. (2018). Logirunner: um Jogo de Tabuleiro como Ferramenta para o Auxílio do Ensino e Aprendizagem de Algoritmos e Lógica de Programação. *RENOTE - Revista Novas Tecnologias na Educação*, 16(1). [GS Search]

Connoly, T. M., Boyle, E. A., Macarthur, E., Hainey, T. & Boyle, J. M. (2012). A systematic literature review of empirical evidence on computer games and serious games. *Computers & Education*, 59(2), 661-686. [GS Search]

CSTA. ACM. CSTA K –12 Computer Science Standards, (2016). Retrieved from <http://k12cs.org/wp-content/uploads/2016/09/K%E2%80%9312-Computer-Science-Framework.pdf>.



- Davis, F. D. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS quarterly*, 319-340. [[GS Search](#)]
- Djaouti, D., Alvarez J., Jessel J. P., & Rampnoux O. (2011). Origins of Serious Games. In: Ma M., Oikonomou A., Jain L. (Eds). *Serious Games and Edutainment Applications*. London: Springer. [[GS Search](#)]
- França, R. & Tedesco, P. (2015). Desafios e oportunidades ao ensino do pensamento computacional na educação básica no Brasil. In: *Anais dos Workshops do Congresso Brasileiro de Informática na Educação*, Maceió, Brasil. [[GS Search](#)]
- Fu, F., Su, R., & Yu, S. (2009). EGameFlow: A scale to measure learners' enjoyment of e-learning games. *Computers & Education*, 52(1), 101-112. [[GS Search](#)]
- Garneli, V., Giannakos, M. N., & Chorianopoulos, K. (2015). Computing education in K-12 schools: A review of the literature. In: Proc. of the IEEE Global Engineering Education Conference, Tallin, Estonia. <https://doi.org/10.1109/EDUCON.2015.7096023> [[GS Search](#)]
- Gomes, T., Melo, J. & Tedesco, P. (2016). Jogos Digitais no Ensino de Conceitos de Programação para Crianças. In: *Anais do XXVII Simpósio Brasileiro de Informática na Educação*, Uberlândia, Brazil. [[GS Search](#)]
- Gresse von Wangenheim, C. & Shull, F. (2009). To Game or Not to Game? *IEEE Software*, 26(2), 92-94. doi: <https://doi.org/10.1109/MS.2009.54> [[GS Search](#)]
- Gresse von Wangenheim, C., Petri, G. & Borgatto, A. F. (2018). *MEEGA+KIDS: A Model for the Evaluation of Educational Games for Computing Education in Secondary School*. Technical Report INCoD/GQS.06.2018.E, INCoD/INE/UFSC, Florianópolis/Brazil. [[GS Search](#)]
- Grover, S. & Pea, R. (2013). Computational Thinking in K-12: A review of the state of the field. *Educational Researcher*, 42(1), 38-43. doi: <https://doi.org/10.3102/0013189X12463051> [[GS Search](#)]
- Guzdial, M. (2008). Education: Paving the way for computational thinking. *Communications of the ACM*, 51(8), 25-27. doi: <https://doi.org/10.1145/1378704.1378713>
- Haddaway, N. R. et al. (2015). The role of Google Scholar in evidence reviews and its applicability to grey literature searching. *PloS one*, 10(9). [[GS Search](#)]
- Herz, J. C. (1997). Joystick nation: how videogames ate our quarters, won our hearts, and rewired our minds. Little, Brown and Company, Boston, MA. [[GS Search](#)]
- International Standard Organization (ISO). (2014). ISO/IEC 25010: Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – System and software quality models, Technical Report. Retrieved from <https://pdfs.semanticscholar.org/57a5/b99ecef9da205e244337c9f4678b5b23d25.pdf>
- Kalelioglu F., Gülbahar Y., & Kukul. V. (2016). A Framework for Computational Thinking Based on a Systematic Research Review. *Baltic Journal of Modern Computing*, 4(3). [[GS Search](#)]
- Kazimoglu, C., Kiernan. M., Bacon, L. & Mackinnon, L. (2012). A Serious Game for Developing Computational Thinking and Learning Introductory Computer Programming. *Procedia - Social and Behavioral Sciences*, 47, 1991-1999. [[GS Search](#)]
- Keller, J. (1987). Development and Use of the ARCS Model of motivational Design. *Journal of Instructional Development*, 10(3), 2-10. doi: <https://doi.org/10.1007/BF02905780> [[GS Search](#)]



Kish, L. (1994). Multipopulation survey designs: five types with seven shared aspects. *International Statistical Review*, 62(2), 167–186. [GS Search]

Kremers, R. *Level Design: Concept, theory, and practice*. A. K. Peters, EUA, 2009. [GS Search]

MEC. Base Nacional Comum Curricular – BNCC. 2018. Retrieved from [http://basenacionalcomum.mec.gov.br/images/BNCC\\_EI\\_EF\\_110518\\_versaofinal\\_site.pdf](http://basenacionalcomum.mec.gov.br/images/BNCC_EI_EF_110518_versaofinal_site.pdf).

Michael, D. & Chen, S.. (2006) *Serious games: games that educate, train, and inform*. Thomson Course Technology, Boston, EUA. [GS Search]

Mioto, F., Petri, G., Gresse von Wangenheim, C., Borgatto, A. F. & Pacheco, L. H. M. (2019). bASES21 – Um Modelo para a Autoavaliação de Habilidades do Século XXI no Contexto do Ensino de Computação na Educação Básica. *Revista Brasileira de Informática na Educação*, 27(1). doi: <http://dx.doi.org/10.5753/rbie.2019.27.01.26> [GS Search]

Mohamed, H. & Jaafar, A. (2010). Development and Potential Analysis of Heuristic Evaluation for Educational Computer Game (PHEG). In: *Proc. of the 5th Int. Conf. on Computer Sciences and Convergence Information Technology*, Seoul, South Korea. [GS Search]

Morrison B. B. & Preston J.A. (2009). Engagement: Gaming throughout the Curriculum. In: *Proc. of the 40th ACM Technical Symposium on Computer Science Education*, Chattanooga, TN. [GS Search]

Nouri, J., Zhang, L., Mannila, L. & Norén, E. (2019). Development of computational thinking, digital competence and 21st century skills when learning programming in K-9. *Education Inquiry*, published online. <https://doi.org/10.1080/20004508.2019.1627844> [GS Search]

O'Brien, H. L. & Toms, E. G. (2010). The Development and Evaluation of a Survey to Measure User Engagement. *Journal of the American Society for Information Science and Technology*, 61(1), 50–69. [GS Search]

Okazaki, M. & Johnson, G. (2013). *Kawaii!: Japan's Culture of Cute*. Munich: Prestel.

Petri G., C. Gresse von Wangenheim & A.F. Borgatto A.F. (2018). MEEGA+, Systematic Model to Evaluate Educational Games. In: Lee N. (eds) *Encyclopedia of Computer Graphics and Games*, Springer. [GS Search]

Petri, G. (2018). A Method for the Evaluation of the Quality of Games for Computing Education. Tese de doutorado (Programa de Pós-Graduação em Ciência da Computação (PPGCC)), Universidade Federal de Santa Catarina (UFSC) [GS Search]

Pfahl, D., Ruhe, G. & Koval, N. (2001). An Experiment for Evaluating the Effectiveness of Using a System Dynamics Simulation Model in Software Project Management Education. In: *Proc. of the Int. Symposium on Software Metrics*, London, GB. doi: <https://doi.org/10.1109/METRIC.2001.915519> [GS Search]

Savi, R., Gresse von Wangenheim, C. & Borgatto, A. F. (2011). A Model for the Evaluation of Educational Games for Teaching Software Engineering. In: *Anais do Simpósio Brasileiro de Engenharia de Software*, São Paulo, Brasil. doi: <https://doi.org/10.1109/SBES.2011.27> [GS Search]

Sociedade Brasileira de Computação. Ensino de Computação na Educação Básica. (2018). Retrieved from <http://www.sbc.org.br/documentos-da-sbc/send/131-curriculos-de-referencia/1177-diretrizes-para-ensino-de-computacao-na-educacao-basica>> .



- Sindre, G. & Moody, D. (2003). Evaluating the Effectiveness of Learning Interventions: an Information Systems Case Study. In: *Proc. of the 11th European Conf. on Information Systems*, Naples, Italy. [[GS Search](#)]
- Singh, J., Kumar Dorairaj, S. & Woods, P. (2007). Learning Computer Programming Using A Board Game – Case Study on C-Jump. In: *Proc. of the Int. Symposium on Information and Communications Technologies*, Kuala Lumpur, Malaysia. [[GS Search](#)]
- Susi, T., Johannesson, M. & Backlund, P. (2007) *Serious Games: an overview*. Technical Report HS-IKI-TR-07-001, University of Skövde, Sweden. [[GS Search](#)]
- Sweetser, P. & Wyeth, P. (2005). GameFlow: a model for evaluating player enjoyment in games. *Computers in Entertainment*, 3(3), 1-24. <http://doi.org/10.1145/1077246.1077253>
- Tullerton, T. (2008) *Game design workshop: A play centric approach to creating innovates games*. 3. ed. Burlington: Morgan Kaufmann, p. 470. 2008. [[GS Search](#)]
- Tsarava, K., Moeller, K. & Ninaus, M. (2018). Training Computational Thinking through board games: The case of Crabs & Turtles. *International Journal of Serious Games*, 5(2), 25-44. doi:<https://doi.org/10.17083/ijsg.v5i2.248> [[GS Search](#)]
- Tullis, T. & Albert, W. (2008). *Measuring the User Experience: Collecting, Analyzing, and Presenting Usability Metrics*. Burlington: Morgan Kaufmann. [[GS Search](#)]
- Unnikrishnan, R., Amrita, N., Muir, A. & Rao, B. (2016). Of Elephants and Nested Loops: How to Introduce Computing to Youth in Rural India. In: *Proc. of the 15th International Conference on Interaction Design and Children*, Manchester, UK, 137–146. doi: <https://doi.org/10.1145/2930674.2930678> [[GS Search](#)]
- Wang, P. S. (2015). *From Computing to Computational Thinking*. 1st ed. New York: Chapman and Hall/CRC. doi: <https://doi.org/10.1201/9781315115320>
- Wiebe, E. N., Lamb, A., Hardy, M. & Sharek, D. (2014). Measuring engagement in video game-based environments: Investigation of the User Engagement Scale. *Computers in Human Behavior*, 32, 123-132. doi: <https://doi.org/10.1016/j.chb.2013.12.001> [[GS Search](#)]
- Wing, J. M. (2006). Computational Thinking. *Theoretical Computer Science*, 49(3). doi:<http://doi.org/10.1145/1118178.1118215>
- Wing, J. M. (2010). Computational Thinking: What and Why? Link Magazine. Retrieved from [https://pdfs.semanticscholar.org/628a/da255c83abfee8693132310cba2ccfaed5a6.pdf?\\_ga=2.77027669.1515728009.1566400560-1938129375.1537444205](https://pdfs.semanticscholar.org/628a/da255c83abfee8693132310cba2ccfaed5a6.pdf?_ga=2.77027669.1515728009.1566400560-1938129375.1537444205)
- Wohlin, C., Runeson, P., Höst, M., Ohlsson, M. C., Regnell, B. & Wesslén, A. (2012). *Experimentation in Software Engineering*. New York: Springer-Verlag Berlin Heidelberg. [[GS Search](#)]
- Yin, R. K. (2017). *Case study research: design and methods* (6th ed.). Thousand Oaks: Sage Publications, Inc. [[GS Search](#)]