


O protagonismo de estudantes da educação básica a partir do desenvolvimento de aplicativos para *smartphone*


Giselle Araújo e Silva de Medeiros
Christiane Gresse von Wangenheim
Jean Carlo Rossa Hauck


Resumo

A importância da aprendizagem do pensamento computacional já é discutida na Educação Básica. O pensamento computacional engloba a resolução de problemas, além de outras habilidades, conhecimento de algoritmos e programação, o que pode ser ensinado por meio do desenvolvimento de aplicativos móveis. Utilizando uma linguagem de programação visual, como *App Inventor*, estudantes da Educação Básica podem desenvolver aplicativos funcionais para *smartphones*, possibilitando a eles o aprendizado de conceitos de computação, de *design* de interface, entre outras habilidades. É adotada como estratégia metodológica a ação computacional, uma abordagem que visa ensinar computação por meio da resolução de problemas na comunidade local. O artigo apresenta um estudo de caso avaliando a experiência de aprendizagem de estudantes no projeto ‘Jovens Tutores de Programação’, a partir da prática pedagógica ‘Faça seu primeiro app’, criada pela Iniciativa Computação na Escola/INCoD/INE/UFSC em 2018, que visa ensinar pensamento computacional a jovens do Ensino Fundamental por meio do desenvolvimento de aplicativos. Além disso, os jovens colaboram ensinando o conhecimento de computação para outros alunos do Ensino Fundamental. Os resultados deste estudo de caso mostram que essa prática pedagógica pode contribuir para o interesse e a motivação dos estudantes, abrangendo temáticas com base na vida deles e da comunidade, e com isso oportunizando o protagonismo dos estudantes. Tais perspectivas tendem a fortalecer a construção do conhecimento e do desenvolvimento de habilidades dos estudantes, incentivando a construção, ao invés de apenas consumo de tecnologia.

Palavras-chave: Pensamento computacional. Aplicativo móvel. Ensino fundamental.

Giselle Araújo e Silva de Medeiros
Rede Municipal de Ensino de
Florianópolis – SC
E-mail: gisellearaujo.ufsc@gmail.com
 <https://orcid.org/0000-0002-7792-4776>

Christiane Gresse von Wangenheim
Universidade Federal de Santa Catarina
(UFSC)
E-mail: c.wangenheim@ufsc.br
 <https://orcid.org/0000-0002-6566-1606>

Jean Carlo Rossa Hauck
Universidade Federal de Santa Catarina
(UFSC)
E-mail: jean.hauck@ufsc.br
 <https://orcid.org/0000-0001-6550-9092>

Recebido em: 02/07/2019
Aprovado em: 18/05/2020



Abstract**The protagonism of K-12 students through the development of smartphone applications**

The importance of learning computational thinking in K-12 is well recognized today. Computational thinking encompasses problem solving as well as knowledge of algorithms and programming, which can be taught through the development of mobile applications. Using a visual block programming language such as App Inventor, middle school students can learn to develop functional applications for smartphones, allowing them to understand and apply concepts of computing, interface design among other skills. Adopting computational action as pedagogical strategy allows to resolve problems in the local community, as part of computing education. In this context, the article presents a case study evaluating the learning experience of the students participating in the project "Young Programming Tutors", through the pedagogic practice "Make your first app", created by the initiative Computação na Escola/INCoD/INE/UFSC in 2018, which aims to teach computational thinking to middle school students through the development of applications. In addition, the students collaborate by teaching computing and knowledge and programming competencies to other students at their school. The results of this case study show that this pedagogical practice can contribute to the interest and motivation of the students, covering subjects with respect to their lives and community. Such perspectives tend to strengthen the construction of knowledge and the development of student skills, encouraging construction rather than just the consumption of technology.

Keywords:

Computational thinking.
Mobile application.
Middle school.

Resumen**El protagonismo de estudiantes de la Enseñanza Básica a partir del desarrollo de aplicaciones para *smartphone***

La importancia de aprender el pensamiento computacional ya se discute en Enseñanza Básica. El pensamiento computacional abarca la resolución de problemas, así como otras habilidades, el conocimiento de algoritmos y la programación que se pueden enseñar a través del desarrollo de aplicaciones móviles. Usando un lenguaje de programación visual como App Inventor, los estudiantes de Enseñanza Básica pueden desarrollar aplicaciones funcionales para teléfonos inteligentes, permitiéndoles aprender conceptos de computación, diseño de interfaces y otras habilidades. Se adopta como acción computacional de estrategia metodológica, un enfoque que apunta a enseñar computación resolviendo problemas en la comunidad local. El artículo presenta un caso de estudio que evalúa la experiencia de aprendizaje de los estudiantes en el proyecto "Jóvenes tutores de la programación", basado en la práctica pedagógica "Haga su primera aplicación", creado por Initiative Computação na Escola/INCoD/INE/UFSC en 2018, que tiene como objetivo enseñar el pensamiento computacional a los jóvenes en la escuela primaria a través del desarrollo de aplicaciones. Además, los jóvenes colaboran en la enseñanza de la alfabetización informática a otros estudiantes de escuela primaria. Los resultados de este estudio de caso muestran que esta práctica pedagógica puede contribuir al interés y la motivación de los estudiantes, cubriendo temas basados en su vida y la comunidad, y proporcionando así el protagonismo de los estudiantes. Tales perspectivas tienden a fortalecer la construcción de conocimiento y el desarrollo de las habilidades de los estudiantes, alentando la construcción en lugar de solo el consumo de tecnología.

Palabras clave:

Pensamiento computacional.
Aplicación móvil.
Educación primaria.

Introdução

A integração das Tecnologias Digitais de Informação e Comunicação (TDIC) na aprendizagem e em ambiente escolar são, atualmente, um dos grandes desafios da educação. Elas representam o conhecimento de novas habilidades a partir de multiletramentos, o que exige uma maior familiaridade de alunos e professores com os recursos digitais (ALMEIDA; VALENTE, 2011). Dentre os caminhos possíveis para alcançar esse conhecimento, Almeida e Valente (2011) apontam a necessidade de práticas focadas em pensamento lógico e computacional, ressaltando que é preciso resgatar o termo “pensamento computacional”, uma discussão relativamente recente, mesmo que o conceito em si já tenha sido apontado anteriormente por Papert (1994), criador da linguagem de programação Logo.¹

O pensamento computacional, definido por Wing (2006), engloba uma abordagem de resolução de problemas que agrega um conjunto de conceitos e tem sido vista como uma habilidade fundamental no século XXI. Isso porque tem foco em demandas mundiais da educação na Era Digital (PÉREZ GÓMEZ, 2015), incluindo capacidades cognitivas derivadas da ciência da computação (RESNICK *et al.*, 2009; GROVER; PEA, 2013; KAFAI; BURKE, 2013). De acordo Almeida e Valente (2019, p. 207), a proposta de Wing (2006) quanto ao pensamento computacional.

[...] despertou o interesse de diversos países em repensar suas políticas e uso das tecnologias digitais de informação e comunicação (TDIC) na educação, e em orientar professores para a realização de práticas que enfatizam a exploração do conceito de programação com foco no desenvolvimento do pensamento lógico e do pensamento computacional.

Para o desenvolvimento do pensamento computacional com alunos da Educação Básica, tipicamente, há o foco no ensino de algoritmos e programação em ambientes de programação visual baseados em blocos, como *Blockly* (GOOGLE, 2019), *Scratch* (MIT, 2016) ou *App Inventor* (MIT, 2019). Esses ambientes, geralmente com uma interface intuitiva, permitem o encaixe de uma peça a partir da correspondência de um código com a sintaxe correta (VALENTE, 2016). Nesse contexto, o ambiente de programação *App Inventor*, por exemplo, permite a programação de aplicações móveis totalmente funcionais para *smartphones* e *tablets*, abordando conceitos de programação como condicionais, eventos, variáveis, entre outros, assim como o *design* da interface do usuário (*user interface*).

Pesquisadores da área baseiam-se no fundamento de que o ensino do pensamento computacional, da programação e do *design* pode contribuir para a formação do estudante do século XXI, preparando-o para lidar com os desafios da Era Digital (PÉREZ GÓMEZ, 2015), a partir da resolução de problemas conectados a questões do mundo real (WANG, 2015; TISSENBAUM; SHELDON; ABELSON, 2019).

Ao conectar-se com vida real dos alunos, podemos ajudá-los desenvolver uma consciência crítica do papel que podem desempenhar em afetar sua comunidade por meio da computação e capacitá-los a ir além de simplesmente aprendendo a codificar. Em vez disso, podemos perguntar a eles o que

eles querem codificar e por que eles querem codificá-lo. (TISSENBAUM; SHELDON; ABELSON, p. 36, 2019).

Este processo pode ocorrer por meio do desenvolvimento de aplicativos móveis em ambientes de programação, como o *App Inventor*, que, com mais de 400 mil usuários ativos atualmente, presentes em 195 países, possibilitou a criação de quase 22 milhões de aplicativos (MIT, 2019) e, por essa razão, tem sido utilizado com bastante sucesso em diferentes iniciativas nos ambientes escolares.

Assim, partindo-se das questões englobando o ensino do pensamento computacional, focado em algoritmos e programação, planejou-se a prática pedagógica ‘Faça seu primeiro app’ dentro do projeto ‘Jovens Tutores de Programação’, desenvolvido de forma sistemática de acordo com o modelo de *design* instrucional ADDIE (BRANCH, 2009). A prática pedagógica, que apresentaremos de maneira mais detalhada a seguir, foi aplicada e avaliada a partir de um estudo de caso desenvolvido com alunos do Ensino Fundamental II de uma escola pública municipal de Florianópolis, SC.

Aplicação da prática pedagógica

A aplicação do projeto ‘Jovens Tutores de Programação’ foi realizada na Escola Básica Municipal Almirante Carvalho, localizada na região continental de Florianópolis, SC, no 2º semestre de 2018. Participaram do projeto dez jovens tutores, dentre os quais cinco meninas e cinco meninos, estudantes do 8º e do 9º ano, com idades entre 13 e 15 anos, selecionados pela equipe da escola.

Para a aplicação deste projeto, além da equipe da Computação na Escola, foi possível contar com o apoio de profissionais da área de tecnologia e de professores da Educação Básica, que se uniram em uma equipe multidisciplinar para ajudar os jovens tutores em dois desafios: desenvolver o seu primeiro app e colaborar com outros alunos do Ensino Fundamental, ensinando o que aprenderam de computação e programação.

Para atingir os objetivos do primeiro desafio, o desenvolvimento de seu próprio aplicativo, foram realizados doze encontros, que seguiram o plano de ensino elaborado pela equipe, conforme apresentado no Quadro 1:

Quadro 1 – Encontros do Projeto Jovens Tutores de Programação

Encontros	Local
Formação com os participantes (Jovens Tutores, profissionais da área de tecnologia e profissionais da escola)	UFSC
Motivação e conceitos básicos de computação e do desenvolvimento de aplicativos; conceitos básicos de computação: algoritmo/ programação	Escola
Processo de desenvolvimento de apps e identificação de problema/ solução	Escola

Análise de contexto e especificação de requisitos	Escola
<i>Design</i> de baixa fidelidade do app e testes	Escola
Programação e teste do protótipo no <i>App Inventor</i> (2 encontros)	Escola
Criação e teste do <i>design</i> visual no <i>App Inventor</i>	Escola
Teste de sistema	Escola
Compartilhando o app	Escola
Preparação para apresentação do app	Escola
Apresentação dos apps desenvolvidos pelos alunos	Empresa <i>Involves</i> , parceira do projeto

Fonte: elaborado pelos autores.

Os encontros aconteceram no contraturno escolar, em uma proposta extracurricular, na qual houve o desenvolvimento do próprio *app*, além de outras atividades envolvendo o ensino de computação. Os encontros possibilitaram aos alunos a aprendizagem de como desenvolver um aplicativo funcional partindo das suas próprias ideias, dentro do amplo desafio de ‘resolver problemas da sociedade’, proposto pela equipe.

O primeiro desses encontros foi realizado na Universidade Federal de Santa Catarina (UFSC), oportunidade em que os estudantes, jovens tutores, participaram de uma formação feita pela equipe da Iniciativa Computação na Escola, juntamente com profissionais da área de tecnologia da empresa apoiadora do projeto. Nessa formação, os alunos aprenderam conceitos básicos de algoritmos e programação e conheceram o ambiente *App Inventor*, no qual programaram o primeiro aplicativo predefinido, que serviu de referência para o posterior processo de desenvolvimento de seus próprios *apps*, adotando práticas ágeis de *design thinking*.

Já para auxiliar no segundo desafio, compartilhar suas aprendizagens, os jovens tutores participaram de três oficinas de programação, o que contribuiu para a sua própria compreensão sobre o processo de ministrar oficinas, atividade que teriam que aplicar aos outros estudantes da escola. A Figura 1 apresenta alguns destes momentos:

Figura 1 – Oficinas aplicadas com estudantes da escola

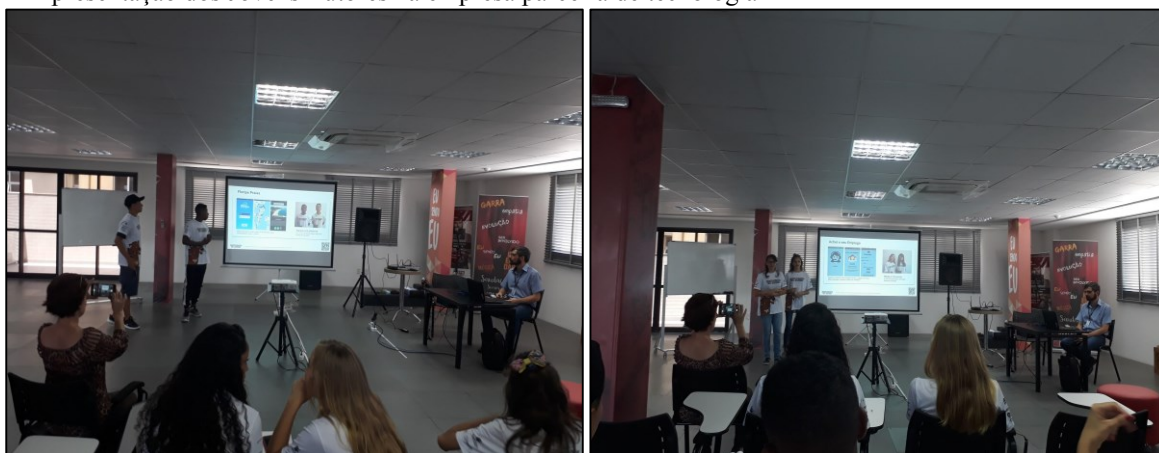


Fonte: acervo particular dos autores.

Nas oficinas que ministraram, os jovens tutores ensinaram aos colegas conceitos de algoritmos utilizando o jogo *SplashCode* (WANGENHEIM *et al.*, 2019), bem como a programação de um aplicativo simples, com o auxílio do *App Inventor*. No total, participaram das oficinas 51 alunos, com faixa etária entre 10 e 14 anos.

Para a finalização do projeto, os jovens tutores apresentaram os seus *apps* na empresa parceira do projeto, conforme se observa na Figura 2. Eles mostraram as ideias desenvolvidas durante a execução do projeto, indicando as razões das suas escolhas e explicitando o problema (que envolve uma questão da comunidade) e a também solução desenvolvida, além da aprendizagem de computação, *design* visual e engenharia de *software*, que ocorreram durante o projeto.

Figura 2 – Apresentação dos Jovens Tutores na empresa parceira de tecnologia



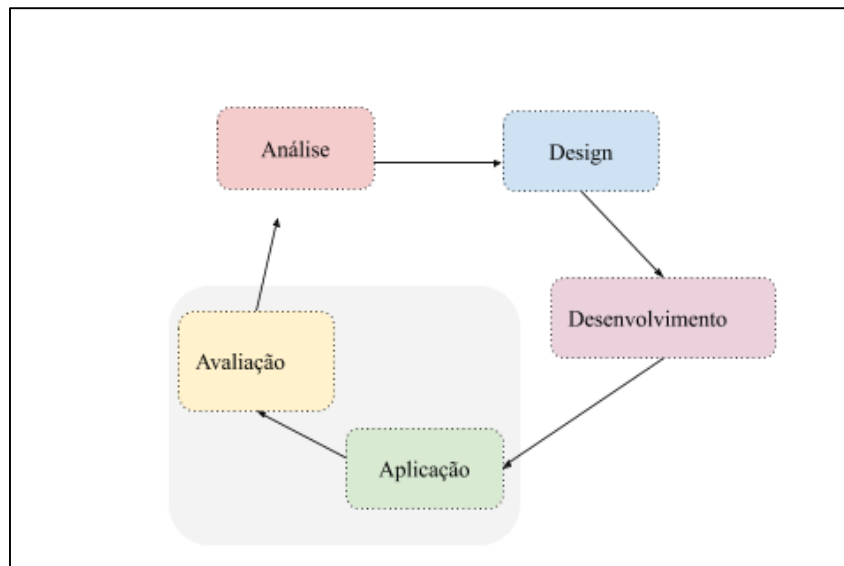
Fonte: acervo particular dos autores.

Metodologia de pesquisa

Nesta pesquisa exploratória, foi adotada uma metodologia multimétodos. De acordo com Gil (2008, p. 27) “[...] as pesquisas exploratórias têm como principal finalidade desenvolver, esclarecer e modificar conceitos e ideias, tendo em vista a formulação de problemas mais precisos ou hipóteses pesquisáveis para estudos posteriores”.

A prática pedagógica foi desenvolvida em obediência ao modelo de *design* instrucional ADDIE (BRANCH, 2009) (Figura 3). Durante a primeira etapa, foi analisado o contexto, tanto em termos do perfil dos alunos quanto do contexto escolar. Durante a etapa do *design*, foram definidos os objetivos de aprendizagem e a estratégia instrucional/pedagógica, documentada no plano de ensino. Na etapa de desenvolvimento, foi criado o material instrucional, com o auxílio do ambiente de programação *App Inventor*.

Figura 3 – Modelo ADDIE



Fonte: adaptado de Branch (2009).

Para a aplicação e avaliação da prática pedagógica, foi realizado um estudo de caso exploratório, conforme as recomendações de Wohlin *et al.* (2012) e Yin (2017), visando à experiência de aprendizagem dos alunos participantes do projeto ‘Jovens Tutores de Programação’, criado pela Iniciativa Computação na Escola, do Instituto Nacional de Convergência Digital (INCoD) do Departamento de Informática e Estatística (INE) da UFSC. A avaliação do estudo de caso foi sistematicamente definida, executada e analisada. Durante a execução, foram coletados dados por meio de questionários, posteriormente analisados com o auxílio de técnicas descritivas de análise.

Desenvolvimento da prática pedagógica ‘Faça seu primeiro app’

Análise do contexto

O público-alvo desta prática pedagógica foram alunos do Ensino Fundamental II, com idades entre 10 e 15 anos. Os alunos tinham acesso a computadores de mesa nas escolas e utilizavam dispositivos móveis como *smartphones*, porém o uso no espaço escolar era restrito a fins educativos.

Há leis em alguns estados brasileiros que, apesar de proibirem o uso do celular na escola, possibilitam sua utilização para fins educativos. O número de não uso é expressivo, o que pode estar atrelado a outras questões que envolvem a infraestrutura, mas também pode ser o reflexo de práticas pedagógicas retrógradas, pouco inovadoras. (MEDEIROS, 2019, p. 31).

Dentro do ambiente escolar, esse acesso acontece nas salas informatizadas, que são equipadas com computadores de mesa e acesso à internet, onde as aulas de tecnologias educacionais acontecem e as práticas pedagógicas são aplicadas, integrando as mídias digitais aos conteúdos curriculares como recursos didáticos, com o uso de *sites* e programas educativos. As atividades desenvolvidas não incluem especificamente o ensino da computação, já que essa área de conhecimento não faz parte do currículo da Educação Básica brasileira. No entanto, em alguns casos, tais práticas são realizadas no contraturno escolar, por meio de projetos.

Assim, com o ensino da computação na Educação Básica, pretende-se desenvolver o pensamento computacional no âmbito de uma prática pedagógica, por meio do desenvolvimento da programação de um aplicativo móvel, seguindo as diretrizes curriculares do ensino de computação na Educação Básica (K-12..., 2017; SBC, 2018).

Definição dos objetivos de aprendizagem

Na prática pedagógica ‘Faça o seu primeiro app’, teve-se como objetivo ensinar o pensamento computacional focado em algoritmos e programação, levando-se em consideração o contexto. Assim, foi abordado principalmente o ensino de algoritmos e programação como parte principal de computação. Foram abordados também conceitos de engenharia de *software* e de usabilidade, áreas importantes para o desenvolvimento de *software*. Os objetivos de aprendizagem dessa prática são apresentados no Quadro 2:

Quadro 2 – Objetivos de aprendizagem

ID	Objetivo de aprendizagem
OA 1	Compreender algoritmos como um conjunto de instruções passo a passo para realizar tarefas.
OA 2	Explique o conceito de um ciclo de vida de <i>software</i> e forneça um exemplo, ilustrando suas fases, incluindo as entregas que são produzidas.
OA 3	Desenvolver artefatos computacionais iterativamente de forma colaborativa, seguindo um cronograma.
OA 4	Identifique e resolva problemas criando sistemas de <i>software</i> interativos.

OA 5	Analisar o contexto de sistemas de <i>software</i> interativo em termos de usuários, tarefas, dispositivos e ambientes de uso.
OA 6	Especificar requisitos de sistemas de <i>software</i> interativos em termos de funcionalidade e usabilidade.
OA 7	Criar protótipos de sistemas de <i>software</i> interativos em diferentes níveis (esboços, baixa fidelidade, alta fidelidade, funcional).
OA 8	Projetar <i>design</i> que combine componentes de <i>hardware</i> e <i>software</i> para coletar e trocar dados (sensores, APIs, etc.).
OA 9	Modelar processos criando e seguindo algoritmos/mapas de navegação para concluir tarefas.
OA 10	Usar fluxogramas, pseudocódigo e/ou mapas de navegação para resolver problemas complexos.
OA 11	Projetar o <i>design</i> visual (cores, tipografia, ícones, imagens, etc.) do sistema de <i>software</i> interativo.
OA 12	Construir sistemas de <i>software</i> interativos que incluam sequenciamento, eventos, condicionais, variáveis, listas e <i>strings</i> usando uma linguagem de programação visual baseada em blocos.
OA 13	Procurar e incorporar o <i>feedback</i> dos membros da equipe e dos usuários para refinar uma solução que atenda às necessidades do usuário.
OA 14	Testar e refinar um sistema de <i>software</i> interativo para funcionalidade e usabilidade.
OA 15	Recomendar melhorias no <i>design</i> de dispositivos de computação, com base nos resultados de verificação e validação.
OA 16	Compartilhar o sistema de <i>software</i> interativo desenvolvido.

Fonte: elaborado pelos autores.

Definição da estratégia pedagógica

A prática pedagógica foi aplicada ao longo de doze encontros semanais, cada qual com duração de aproximadamente 3 horas, além de atividades assíncronas via plataforma *moodle*. Foram etapas interessantes para os participantes, com encontros dentro e fora da escola, o que possibilitou uma ampliação da compreensão sobre outros ambientes educativos de aprendizagem. Isso incluiu encontros na Universidade Federal de Santa Catarina (UFSC) e na empresa parceira, despertando a curiosidade dos alunos sobre futuras áreas de atuação, o que pode vir a contribuir com seus projetos de vida.

Com o objetivo de proporcionar aos alunos uma experiência de aprendizagem motivadora, foi utilizada como abordagem pedagógica a ação computacional, uma proposta que busca a solução de tecnologias que tenham impacto social na vida dos alunos (TISSENBAUM; SHELDON; ABELSON, 2019). Foram adotados vários métodos instrucionais, enfatizando-se a aplicação prática dos conceitos abordados com atividades práticas de desenvolvimento de aplicativos móveis:

- Instrução direta: apresentação de conteúdos e informações pelos professores, por meio de passo a passo e de exemplos;
- Instrução indireta: os alunos têm maior autonomia para tomada de decisões e resolução de

problemas. O professor é um facilitador, que media o processo de aprendizagem, dando *feedbacks* aos alunos;

- Instrução interativa: estrutura de dinâmica de grupo, que possibilita trocas entre professores, alunos e colegas;
- Aprendizagem experimental: tanto dentro da sala de aula como fora dela. Na atualidade, pode-se indicar o contato via *moodle* e por aplicativos;
- Estudos independentes: encoraja os alunos a planejar e colocar em prática o aprendizado desenvolvido na parte da preparação para apresentação dos resultados.

No que diz respeito às estratégias pedagógicas aplicadas neste projeto, elas foram definidas a partir dos objetivos de aprendizagem previamente determinados pela equipe multidisciplinar. Dentre as estratégias utilizadas, destaca-se a compreensão do conceito de algoritmos de forma lúdica, por meio do jogo de tabuleiro *SplashCode* (WANGENHEIM *et al.*, 2019), desenvolvido pela Iniciativa Computação na Escola INCoD/INE/UFSC, assim como atividades complementares, realizadas como tarefas de casa; a realização de oficinas de programação com outras turmas da escola; o compartilhamento do conhecimento aprendido pelos jovens tutores; e a socialização dos aplicativos desenvolvidos pelos próprios alunos. A Figura 4, abaixo, ilustra o resultado de algumas dessas atividades:

Figura 4 – Exemplos de materiais didáticos desenvolvidos



Jogo educacional *SplashCode*



Slides





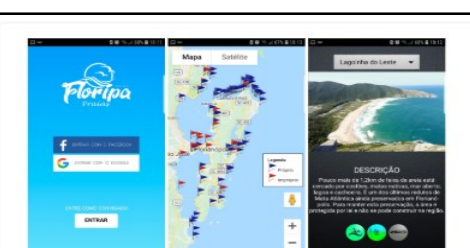

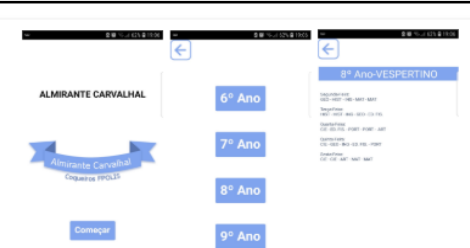
Tutoriais



Tarefa de casa no ambiente *moodle*

Fonte: Pinheiro, (2019), Missfeldt Filho (2019) e Wangenheim *et al.* (2019).

Quadro 3 – Apps desenvolvidos pelos jovens tutores durante a aplicação do projeto² (continua)

Apps desenvolvidos	Slides de apresentação
<p>Achei o seu Emprego</p>	 <p>Auxilia a encontrar o seu emprego. Permite o cadastro de vagas pelas empresas e também a busca por empregos.</p>
<p>AppPhone</p>	 <p>Ajuda a encontrar telefones úteis, especialmente em momentos de emergência.</p>
<p>Floripa Praias</p>	 <p>Exibe informações sobre praias de Florianópolis como balneabilidade, acesso e trilhas.</p>
<p>HealthyPlants</p>	 <p>Mostra informações úteis para ajudar a cuidar de plantas em casa.</p>
<p>InfoCarvalhoal</p>	 <p>Apresenta informações úteis sobre a Escola Básica Municipal Almirante Carvalho: turmas, alunos, horários e professores.</p>

Fonte: elaborado pelos autores.

Avaliação da prática pedagógica

O objetivo deste estudo de caso é avaliar a experiência de aprendizagem dos alunos a partir do projeto ‘Jovem Tutores de Programação’. Buscou-se com isso compreender os interesses e as motivações para aprendizagem escolar, considerando-se também a aprendizagem significativa, que “[...] é uma aprendizagem em que somos capazes de atribuir significado ao conteúdo aprendido” (ZABALA, 2002, p. 97). Zabala (2002) ressalta ainda que, sem entender o significado, não se pode dizer que se aprendeu um conceito ou um princípio.

Definição da Avaliação

A medição da avaliação adotou a abordagem *Goal/Question/Metric* (GQM) (BASILI; CALDIERA; ROMBACH, 1994), que prevê a decomposição da pergunta de pesquisa em medidas e guia o desenvolvimento de instrumentos de coleta de dados.

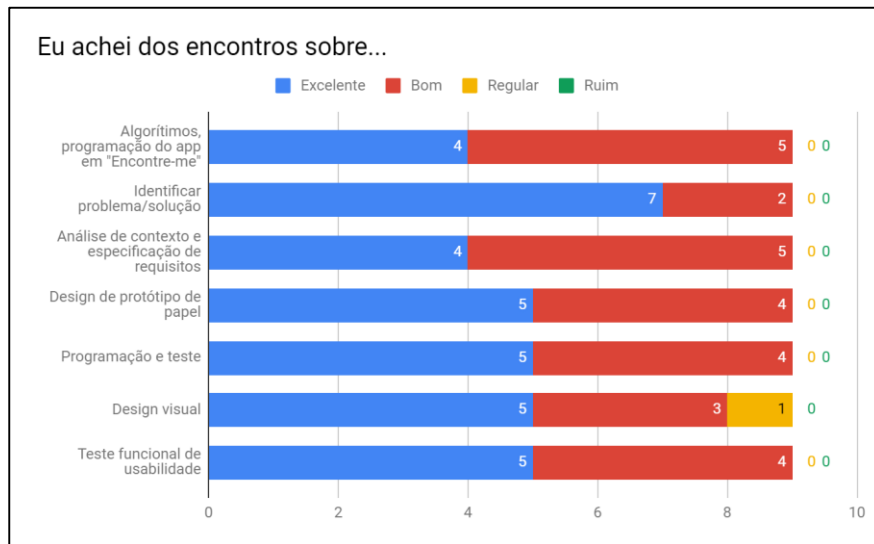
Dessa forma, partiu-se da seguinte **pergunta de pesquisa**: práticas pedagógicas de ensino de computação por meio do desenvolvimento de aplicativos móveis contribuem para o interesse e a motivação de alunos no contexto escolar?

Para desenvolver essa análise, a avaliação foi realizada de forma qualitativa e quantitativa no nível de reação e aprendizagem, conforme o modelo de Kirkpatrick e Kirkpatrick (2006). Foram coletados dados por meio de questionários, aplicados aos alunos, aos profissionais da empresa parceira do projeto e aos professores da escola, além de observações e da avaliação dos projetos dos alunos.

Dados da pesquisa

Para o recorte deste estudo, com o propósito de apresentar a avaliação dos alunos a partir da sua experiência como participantes do projeto ‘Jovens Tutores de Programação’, as respostas nos questionários demonstram que foi uma experiência positiva para os estudantes, como demonstrado no Gráfico 1, cujas respostas, na maior parte, exibem avaliações entre ‘bom’ e ‘excelente’:

Gráfico 1 – Avaliação dos encontros



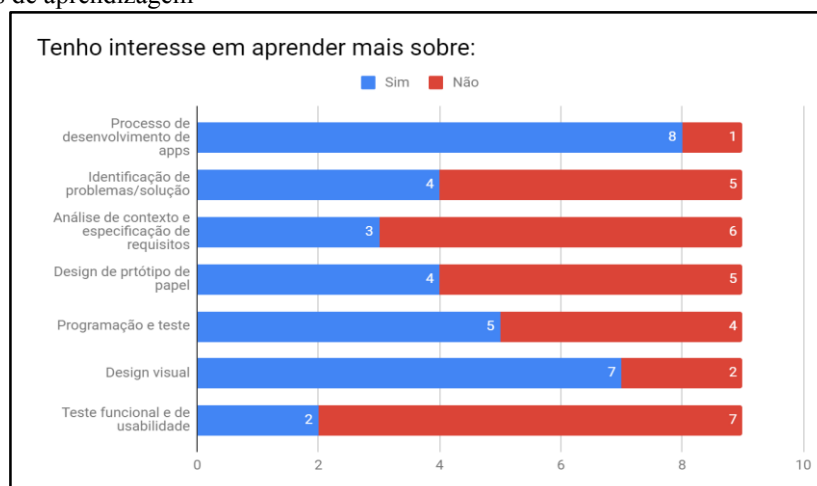
Fonte: elaborado pelos autores.

O encontro sobre identificação do problema/solução, que engloba o processo de desenvolvimento de apps sobre temáticas escolhidas pelos alunos, destaca-se com o maior número de avaliações 'excelentes'. Pode-se inferir que o desenvolvimento de um aplicativo do interesse dos alunos é um fator bastante significativo para eles, opinião corroborada pela literatura especializada da área:

Para conseguir que os alunos se interessem é preciso que os objetivos de saber, realizar, informar-se, e aprofundar sejam uma consequência de interesses detectados; que eles possam saber sempre o que se pretende nas atividades que realizam e que sintam que o que fazem satisfaz alguma necessidade. Mas para isso é indispensável que os meninos e meninas tenham a oportunidade de expressar suas próprias ideias e, a partir delas, convém potencializar as condições que lhe permitam revisar a fundo estas ideias e ampliar as experiências com outras novas, fazendo com que deem conta, também de suas limitações, situando-os em condição de modificá-las se for necessário, ao mesmo tempo que se buscam outras alternativas. (ZABALA, 1998, p. 94).

Quando questionados sobre o que mais lhe despertava interesse em aprender (Gráfico 2), os processos de desenvolvimento de aplicativos e *design* visual foram a opção de maior destaque positivo por parte dos alunos:

Gráfico 2 – Preferências de aprendizagem



Fonte: elaborado pelos autores.

Já os testes tiveram quase 100% de rejeição, pois se trata de uma atividade desgastante, devido à repetição de ações e também à infraestrutura.

No Quadro 4, são apresentados os comentários de uma autoavaliação descritiva simples, com a qual se buscou saber do que os alunos mais e menos gostaram ao fazer o app:

Quadro 4 – O que mais/menos gostei?

O que mais gostei de fazer no app foi?	O que menos gostei de fazer no app foi?
<i>Design</i> visual, paleta de cores e tal (4)	Protótipo de papel
Tudo (3)	Fazer questionário
Programar	Pesquisar plantas e descrições (2)
Aprender a fazer um app	Nada (5)

Fonte: elaborado pelos autores.

Quando perguntados sobre o que mais gostaram de fazer durante o processo de elaboração do aplicativo, a maior parte dos alunos optou pelo *design* visual do aplicativo (Quadro 4). Pelas respostas, observa-se também que, de forma geral, foi uma experiência muito positiva, com três alunos indicando que gostaram de tudo e cinco dizendo que não houve nada de que não gostaram. Duas respostas indicaram que o que menos gostaram de fazer foram as pesquisas específicas sobre a temática plantas, já que um dos apps possui uma miniciclopédia de plantas, algo específico deste app.

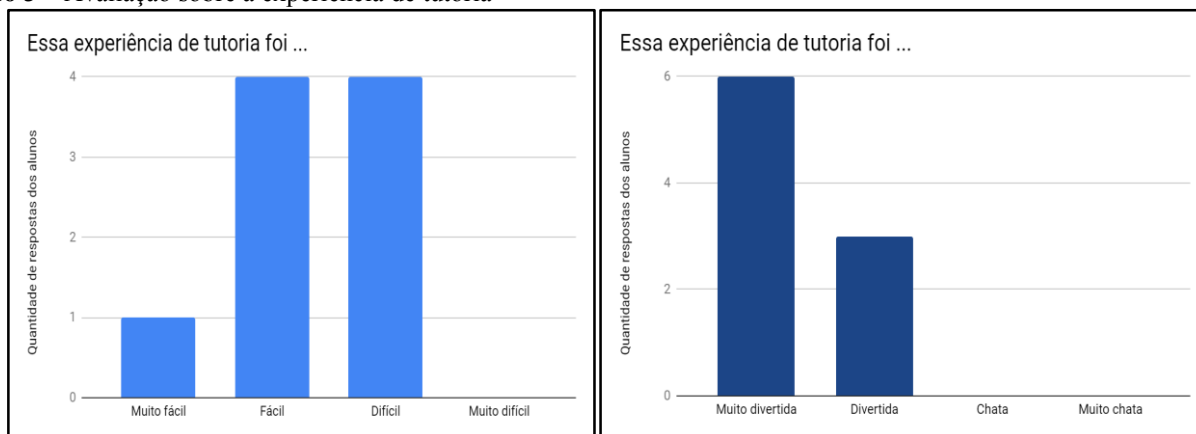
Quadro 5 – O que achou dessa experiência?

O que você achou dessa experiência?
Muito legal, adquirimos muitos conhecimentos
Legal
Uma experiência que nunca foi esquecer. Também foi muito top as aulas e oficinas
Achei bem legal e divertida
Achei bem legal e divertida a experiência de fazer um aplicativo
Muito legal
Muito bacana, gostei demais
Super legal
Não identificado

Fonte: elaborado pelos autores.

A partir das respostas apresentadas, evidencia-se que a experiência foi considerada positiva para os alunos, contribuindo para a construção de conhecimentos de maneira aproximada aos interesses por tecnologias digitais.

De acordo com as respostas dos questionários, a experiência de ser um jovem tutor apresentou aspectos significativos para os estudantes (Gráfico 3):

Gráfico 3 – Avaliação sobre a experiência de tutoria

Fonte: elaborado pelos autores.

No que diz respeito à experiência de tutoria, os itens ‘fácil’ e ‘difícil’ obtiveram a mesma quantidade de respostas (4) (Gráfico 3). Tais respostas são totalmente compreensíveis, já que explicar um conteúdo para outros estudantes exige uma diversidade de habilidades comunicacionais, além de um bom domínio de competências englobando as tecnologias, que é algo ainda em processo no Ensino Fundamental. Quanto ao fato de a tutoria ter sido agradável ou não, houve só respostas positivas, todos concordaram que foi uma experiência prazerosa.

Dessa forma, os dados indicam bons resultados quanto à pergunta de pesquisa, demonstrando que a prática pedagógica de ensino de computação por meio do desenvolvimento de aplicativos móveis, conforme planejado no projeto ‘Jovens Tutores de Programação’, contribuiu de maneira positiva para o interesse e a motivação dos alunos do Ensino Fundamental. Pode-se dizer que estes aspectos positivos colaboraram como um primeiro passo em direção à construção do conhecimento, e que o desenvolvimento de práticas pedagógicas com atenção aos interesses e às motivações dos alunos favorece o protagonismo dos estudantes.

Considerações finais

O estudo de caso apresenta indícios de que a prática pedagógica ‘Faça seu primeiro app’ demonstrou, de forma positiva, por meio da avaliação da experiência dos alunos, que o projeto ‘Jovens Tutores de Programação’ pode contribuir para a aprendizagem de computação. A adoção da abordagem de ação computacional, com a qual se criou o desafio de desenvolver um aplicativo móvel com vistas a solucionar um problema da sociedade, demonstrou contribuir para uma aprendizagem significativa dos alunos na Era Digital. Tais perspectivas tendem a fortalecer a construção do conhecimento e o desenvolvimento das habilidades dos alunos, incentivando a construção de soluções tecnológicas, ao invés de apenas o consumo de tecnologia.

Aliar práticas pedagógicas à resolução de problemas do dia a dia pode vir a contribuir para o desenvolvimento da pesquisa, por meio de conteúdos curriculares de abordagem interdisciplinar, colaborando com o contexto escolar. E, na Era Digital, com a presença maciça de dispositivos móveis na vida das pessoas, propostas pedagógicas com *smartphones* e *tablets* são alternativas para o desenvolvimento de atividades com tecnologias digitais.

Nesse sentido, a utilização do *App Inventor*, ambiente de programação visual baseado em blocos, facilitou a aproximação dos alunos com o ensino de computação, despertando o seu interesse ainda no Ensino Fundamental, a partir do desenvolvimento de aplicativos funcionais.

Dessa forma, a aplicação trouxe resultados significativos, dando indícios de que potencialidades podem vir a ser desenvolvidas no ensino de computação na Educação Básica. Entretanto, há carência de pesquisas nesta área, tanto teóricas quanto práticas, principalmente no Brasil, para fins de comparação e de análise.

No entanto, pode-se dizer, pelo projeto ‘Jovens Tutores de Programação’, que práticas pedagógicas englobando o ensino de computação em contexto escolar tendem a contribuir para o interesse e a motivação dos alunos e, conseqüentemente, podem facilitar o desenvolvimento de competências e habilidades para o século XXI.

Agradecimentos

Este trabalho foi apoiado pelo Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), entidade do governo brasileiro voltada ao desenvolvimento científico e tecnológico, e a pela empresa Involves Soluções Tecnológicas S. A., patrocinadora do projeto Jovens Tutores.

Notas

¹ A linguagem *Logo* foi criada por volta de 1960, buscando agregar a interação entre aluno e computador na resolução de problemas.

² Os apps estão disponíveis na Google Play Store: https://play.google.com/store?hl=pt_BR. Acesso em: 20 set. 2019.

Referências

ALMEIDA, Maria Elisabeth Bianconcini; VALENTE, José Armando. **Tecnologias e Currículo: trajetórias convergentes ou divergentes?** São Paulo: Paulus, 2011.

ALMEIDA, Maria Elisabeth Bianconcini; VALENTE, José Armando. Pensamento computacional nas políticas e nas práticas em alguns países. **Revista Observatório**, Palmas, v. 5, n. 1, p. 202-242, 2019.

BASIL, Victor R.; CALDIERA, Gianluigi; ROMBACH, H. Dieter. **The goal question metric approach, encyclopedia of software engineering**. [S. l.: s. n.], 1994. Disponível em: <https://bit.ly/31UZs5V> Acesso em: 20 set. 2019.

BRANCH, R. **Instructional design: The ADDIE approach** (2nd ed.). Nova York: Springer. 2009.

K-12 Computer Science Standards. Computer Science Teachers Association, 2017. Disponível em: <http://www.k12cs.org>. Acesso em: 2 jul. 2019.

GIL, Antonio Carlos. **Métodos e técnicas de pesquisa social**. 6. ed. São Paulo: Atlas, 2008.

GOOGLE. Blockly. Versão 2.0. [Mountain View]: Google, 2019. Disponível: <https://developers.google.com/blockly/>. Acesso em: 2 de junho de 2019.

GROVER, Suchi; PEA, Roy. Computational thinking in K-12: A review of the state of the field. **Educational Researcher**, [S. l.], v. 42, n. 1, p. 38-43, jan. 2013. DOI: <https://doi.org/10.3102/0013189X12463051>.

KAFAI, YasminB.; BURKE, Quinn. Computer programming goes back to school. **Phi Delta Kappan**, [S. l.], v. 95, n. 1, p. 61-65, sep. 2013. ISSN: 0031-7217. DOI: <https://doi.org/10.1177/003172171309500111>.

KIRKPATRICK, Donald L.; KIRKPATRICK, James D. **Evaluating training programs: the four levels**. 3. ed. [S. l.]: Berrett-Koehler Publishers, 2006. Disponível em: <https://bit.ly/3bmJzrX>. Acesso em: 20 set. 2019.

K-12 Computer Science Standards. Computer Science Teachers Association. **K12cs**, [S. l.], 2017. 2017. Disponível em: <http://www.k12cs.org>. Acesso em: 2 jul. 2019.

MASSACHUSETTS INSTITUTE OF TECHNOLOGY – MIT. **Scratch**. Version 2.0. [Cambridge]: MIT, 2016. Disponível em: <http://appinventor.mit.edu/explore/about-us.html/>. Acesso em: 8 de maio de 2019.

MASSACHUSETTS INSTITUTE OF TECHNOLOGY – MIT. **App Inventor**. Version 4.0. [Cambridge]: MIT, 2019. Disponível em: <http://appinventor.mit.edu/explore/about-us.html/>. Acesso em: 8 de maio de 2019.

MEDEIROS, Giselle Araújo e Silva. **Práticas pedagógicas com o desenvolvimento de aplicativos: dispositivos móveis e aprendizagem significativa para Ensino Fundamental**. 2019. 198 f. Dissertação (Mestrado em Educação) – Centro de Ciências da Educação, Florianópolis, 2019.

MISSFELDT FILHO, Raul. **Desenvolvimento de uma unidade instrucional para ensinar o desenvolvimento de apps no ensino fundamental com o app inventor**. 2019. Trabalho de Conclusão de Curso (Graduação em Ciência da Computação) – Universidade Federal de Santa Catarina, Florianópolis, 2019.

PAPERT, Seymour. **A máquina das crianças: repensando a escola na era da informática**. Porto Alegre: Artes Médicas, 1994.

PÉREZ GÓMEZ, Angel. **Educação na Era Digital: a escola educativa**. Porto Alegre: Penso, 2015.

PINHEIRO, Fernando da Cruz. **Modelo instrucional para o ensino de Engenharia de Software e Usabilidade voltado à Educação Básica**. 2019. 231 f. Dissertação (Mestrado em Ciência da Computação) – Universidade Federal de Santa Catarina, Florianópolis, 2019.

RESNICK, Mitchel *et al.* Scratch: Programming for all. **Communications of the ACM**, v. 52, n. 11, p. 60-67, dec. 2009. DOI: <https://doi.org/10.1145/1592761.1592779>.

SOCIEDADE BRASILEIRA DE COMPUTAÇÃO. Ensino de Computação na Educação Básica. [S. l.: s. n.], 2018. Disponível em: <https://bit.ly/3jHeuC1>. Acessado em: 18 maio 2019.

TISSENBAUM, Michael, SHELDON, Josh e ABELSON, Hal. From computational thinking to computational action: envisioning computing education that both teaches and empowers. **Communications of the ACM**, [S. l.], v. 62, n. 3, p. 34-36, feb. 2019. DOI: <https://doi.org/10.1145/3265747>.

UNIVERSIDADE FEDERAL DE SANTA CATARINA. Departamento de Informática e Estatística. Iniciativa Computação na Escola. **Computação na escola**, [S. l.], 2019. Disponível em: <http://www.computacaonaescola.ufsc.br> Acessado em 6 de maio de 2019.

VALENTE, José. Armando. Integração do Pensamento Computacional no currículo da Educação Básica: diferentes estratégias usadas e questões de formação de professores e avaliação do aluno. **Revista e-Curriculum**, São Paulo, v. 14, n. 3, p. 864-897, jul./set. 2016. Disponível em: <https://bit.ly/2GRBiBi> Acesso em: 20 set. 2019.

WANG, Paul S. **From computing to computational thinking**. 1st ed. New York: Chapman and Hall, 2015.

WANGENHEIM, Christiane Gresse von; MEDEIROS Giselle, MISSFELDT FILHO, Raul; PETRI, Giani; PINHEIRO, Fernando da Cruz, FERREIRA Mirian Nathalie Fortuna; HAUCK Jean Carlos Rossa. SplashCode – A Board Game for Learning an Understanding of Algorithms in Middle School. *Informatics in Education*, accepted for publication. **Informatics in Education**, [S. l.], V. 18, n. 2, p. 259-280, 2019. ISSN: 2335-8971. DOI: <https://doi.org/10.15388/infedu.2019.12>.

WOHLIN, Claes *et al.* **Experimentation in Software Engineering**. Berlin: Springer-Verlag Berlin Heidelberg, 2012.

WING, Jeannette M. Computational thinking. **Communications of the ACM**, [S. l.], 49, n. 3, p. 33-35, 2006. Disponível em: <https://bit.ly/2R4NbFP>. Acesso em: 20 set. 2019.

YIN, Robert K. **Case study research: design and methods**. 6th ed. Thousand Oaks: Sage Publications, 2017.

ZABALA, Antoni. **A prática educativa: como ensinar**. Porto Alegre: Artmed, 1998.

ZABALA, Antoni. **Enfoque globalizador e pensamento complexo: uma proposta para o currículo escolar**. Porto Alegre: Artmed, 2002.