



UNIVERSIDADE DO ESTADO DA BAHIA
DEPARTAMENTO DE CIÊNCIAS EXATAS E DA TERRA
CURSO DE GRADUAÇÃO EM SISTEMAS DE INFORMAÇÃO

ISABEL VALDERRAMA ATTA

**INTERVENÇÕES NA EDUCAÇÃO BÁSICA COM BASE NO
PENSAMENTO COMPUTACIONAL: O AMBIENTE #INCLUDE**

SALVADOR

2020

ISABEL VALDERRAMA ATTA

INTERVENÇÕES NA EDUCAÇÃO BÁSICA COM BASE NO
PENSAMENTO COMPUTACIONAL: O AMBIENTE #INCLUDE

Monografia apresentada ao curso de Sistemas de Informação do Departamento de Ciências Exatas e da Terra da Universidade do Estado da Bahia UNEB, como requisito à obtenção do grau de Bacharel em Sistemas de Informação. Área de Concentração: Ciência da Computação.

Orientador: Antonio Carlos Fontes Atta

SALVADOR
2020

FICHA CATALOGRÁFICA
Sistema de Bibliotecas da UNEB
Dados fornecidos pelo autor

A883i

Atta, Isabel Valderrama

Intervenções na Educação Básica com base no Pensamento Computacional: O Ambiente #INCLUDE / Isabel Valderrama Atta.-- Salvador, 2020.

53 fls.

Orientador(a): Prof. Antonio Carlos Fontes Atta.

Inclui Referências

TCC (Graduação - Sistemas de Informação) - Universidade do Estado da Bahia. Departamento de Ciências Exatas e da Terra. Câmpus I. 2020.

1.Pensamento Computacional. 2.Educação Básica.
3.Pesquisa-Aplicação.

CDD: 003

Agradecimentos

Agradeço aos meus pais por todo amor, dedicação e apoio conferido não apenas ao longo do desenvolvimento deste trabalho, mas por toda minha vida. Seus ensinamentos vão muito além de questões técnicas, sou grata por todas as oportunidades de discussão e crescimento.

Agradeço também aos meus amigos Fábio Ramos, Samuel Carvalho, Tarcio Almeida, Wellington Correia e Juliana Santana por todo o companheirismo e alegrias proporcionadas no decurso dos últimos anos.

Agradeço à Universidade do Estado da Bahia por possibilitar o acesso a uma educação pública de elevada qualidade, fomentando o desenvolvimento científico a partir de uma formação abrangente e emancipadora.

Agradeço às especialistas que avaliaram o #INCLUDE, sem as suas importantes contribuições o trabalho não estaria completo. Essa etapa me proporcionou a experiência de realizar uma pesquisa científica plena, sou muito grata pela oportunidade.

Por fim, gostaria de agradecer a todas as mulheres da computação por suas contribuições para a ciência. Desde Ada Lovelace até todas as professoras do curso de Sistemas de Informação da Universidade do Estado da Bahia, deixo registrada minha sincera gratidão por incentivar e inspirar meninas a ocuparem espaços historicamente masculinos a partir da promoção do conhecimento científico.

“A alegria não chega apenas no encontro do achado, mas faz parte do processo da busca. E ensinar e aprender não pode dar-se fora da procura, fora da boniteza e da alegria.”

Paulo Freire

Resumo

Os meios digitais influenciam nossas vidas com uma relevância progressiva desde o seu surgimento. No processo evolutivo humano mudamos a forma de nos relacionar com o mundo a partir do desenvolvimento das redes de conexões suportadas pelas tecnologias de comunicação. Essas transformações não deixam de notar-se significativas também na área da Educação, tendo como um forte conceito em evidência o pensamento computacional. Este trabalho está inserido na temática de compreensão do potencial do pensamento computacional à aplicação na Educação Básica, tendo como objeto de estudo a concepção de uma solução pedagogicamente orientada e idealizada para promover intervenções com foco no desenvolvimento do pensamento computacional na Rede Pública de ensino, notadamente carente de iniciativas nessa área. Com base nas investigações e achados científicos disponíveis sobre o tema, e apoiada na metodologia de pesquisa Pesquisa-aplicação (conhecida também como *Design Based Research* – DBR), um ambiente de intervenção educacional foi projetada e implementada conforme os ciclos avaliativos dessa metodologia – o ambiente #INCLUDE. Os resultados medidos qualitativamente por meio da participação direta de especialistas na área de Educação e de professores com atuação e experiência na Educação Básica demonstram que a proposta é potencialmente promotora de intervenções reais, planejadas e desenvolvidas por professores das etapas dos Ensinos Fundamentais e Médio a partir do emprego efetivo das Tecnologias de Informação e Comunicação – TIC.

Palavras-chave: Pensamento Computacional. Educação Básica. Pesquisa-aplicação.

Abstract

Digital media influences our lives with progressive relevance since its emergence. In the human evolution process we have modified the way of relating to world and each other through networks supported by communications technologies. The technological development shows significance in the field of education as well, through the concept of computational thinking. This work is inserted in the theme of understanding the potential of computational thinking applied in K-12, investigating as the object of study the conception of a pedagogically oriented and idealized solution to promote interventions in Brazilian's Public Education System, that, notably, lacks initiatives in this area. Based on examination of the scientific results available on the topic, supported by the research methodology Design Based Research - DBR , an educational intervention environment was designed and implemented according to the evaluation cycles of this methodology – the #INCLUDE environment. The results measured qualitatively through the direct involvement of specialists in Education and experienced K-12 teachers demonstrates that the proposal is potentially a promoter of real interventions, planned and developed by teachers from the Elementary and High School stages, with effective use of Information and Communication Technologies - ICT.

Keywords: Computational Thinking. K-12. Design Based Research.

Lista de figuras

Figura 1 – Refinamento de problemas, soluções, métodos e princípios de elaboração.....	21
Figura 2 – Visão geral da solução.....	26
Figura 3 – Tela inicial do Professor.....	27
Figura 4 – Tela referente a criação da atividade.....	28
Figura 5 – Tela do Painel de Controle para o gerenciamento de uma atividade.....	29
Figura 6 – Tela do aluno em que será realizada resolução da atividade.....	30
Figura 7 – Modelo de Entidade e Relacionamento do ambiente #INCLUDE.....	34
Figura 8 – Versão Final da Tela Inicial do Professor.....	35
Figura 9 – Tela de cadastro do professor.....	36
Figura 10 – Módulo de atividade - visão do aluno.....	37

Lista de abreviaturas e siglas

API	<i>Application Programming Interface</i>
BNCC	Base Nacional Comum Curricular
DBR	<i>Design-Based Research</i>
GEOTEC	Grupo de Geotecnologias, Educação e Contemporaneidade
PPGEduC	Programa de Pós-graduação em Educação e Contemporaneidade da UNEB
SBC	Sociedade Brasileira de Computação
SGBD	Sistema Gerenciador de Banco de Dados
SQL	<i>Structured Query Language</i>
STEAM	<i>Science, Technology, Engineering, Arts and Mathematics</i>
UNEB	Universidade do Estado da Bahia

Sumário

1	Introdução	10
2	O Pensamento Computacional e suas abordagens na atualidade.....	13
2.1	Iniciativas mundiais relativas ao ensino do pensamento computacional.....	14
2.2	Ferramentas de software com foco no ensino do pensamento computacional.	16
2.3	Trabalhos Correlatos.....	19
3	Objeto de Estudo e Metodologia de Pesquisa	20
4	Desenvolvimento da solução guiado pela Pesquisa-aplicação.....	24
4.1	Análise e exploração do problema.....	24
4.2	Desenvolvimento da solução utilizando Princípios de Design	24
4.3	Implementação e avaliação em ciclos iterativos.....	25
4.3.1	Descrição do Protótipo.....	26
4.3.2	Entrevista com especialista para avaliação do protótipo	30
4.3.3	Implementação da solução.....	32
4.3.4	Avaliação de #INCLUDE – Segundo ciclo avaliativo	38
4.4	Princípio de Design Resultante.....	42
5	Considerações finais	43
6	Referências	45
	Apêndices	48

1 INTRODUÇÃO

Na década de 60 Seymour Papert desenvolveu linguagem de programação Logo (VALENTE, 2016), com o intuito de ensinar crianças conceitos de geometria a partir de comandos direcionados a uma tartaruga, representada digitalmente por um pequeno triângulo que se movia na tela de um computador e que desenhava uma linha contínua como rastro por onde passava (MORAIS, BASSO e FAGUNDES, 2017). Essa proposta constituiu um dos primeiros exemplos de associação das tecnologias da Computação com a Educação. Apesar dessas iniciativas, o discurso a respeito da percepção dos fundamentos por trás dos processos computacionais envolvidos, por muito tempo, foi substituído por intervenções que pretendiam utilizar o computador apenas como um meio para pesquisas e produção de conteúdo. Escolas com salas computadorizadas, acesso a redes sem fio e aulas que versam sobre letramento digital tornaram-se uma realidade inescapável do século 21. Contudo, esse esforço sem uma associação com as habilidades relacionadas aos que projetam as máquinas e as aplicações utilizadas extensivamente por estudantes não mais condiz com as demandas de formação multidisciplinar valorizadas pelo mercado de trabalho.

O conceito de pensamento computacional cunhado por Jeannette Wing (2006) dialoga diretamente com essa necessidade. A *Royal Society* (2012) expressa de forma sucinta a definição da ideia de Wing como “O processo de reconhecer os aspectos computacionais no mundo que nos rodeia, e aplicar ferramentas e técnicas da ciência da computação para compreender e raciocinar sobre os sistemas e processos naturais” (*apud* BOMBASAR, SANTIAGO, *et al.*, 2015). Aprofundando essa definição, é possível destrinchar o “processo de reconhecer aspectos computacionais no mundo que nos rodeia” como habilidades relacionadas à capacidade de abstração, resolução de problemas a partir do pensamento algorítmico, análise de dados, entre outras, que apresentam-se indispensáveis para a formação de estudantes preparados para assumir riscos calculados, engajados no aprendizado experimental, resilientes na solução de problemas complexos, colaborativos e aptos a resolverem situações de forma criativa, abordagem vigente na filosofia STEAM (*Science, Technology, Engineering, Arts and Mathematics*), como forma de alcançar os objetivos de formação para a economia deste século (EDUCATION CLOSET, 2014).

Desenvolver a habilidade do pensamento computacional na solução de problemas e desafios que se apresentam no dia a dia, tanto no âmbito pessoal quanto profissional, é, portanto, um diferencial que representará uma vantagem cognitiva, especialmente para alunos de Rede Pública de ensino que não tem acesso a ferramentas e métodos que fomentam tais habilidades. Essa afirmação é corroborada pelos investimentos amplamente anunciados por iniciativas privadas do setor da Educação, que aplicam recursos, em proporções cada vez maiores, em tecnologias que visam proporcionar formação específica na área de programação de computadores, a exemplo das classes e competições de robótica. Na outra ponta, a educação na Rede Pública carece de propostas com o mesmo viés, viáveis economicamente, que repercutam no desenvolvimento do pensamento computacional entre os alunos dessas instituições, já que os recursos financeiros nesse setor costumam ser escassos e prioritariamente aplicados na formação e atendimento ao currículo básico.

Portanto, este trabalho se propõe a desenvolver uma solução pautada em uma metodologia para aplicação pedagógica na Educação Básica, passível de execução na Rede Pública de ensino, a partir de um conteúdo de formação que propicie uma abordagem efetiva para a produção de conhecimento em/a partir do pensamento computacional. Adicionalmente, o trabalho fundamenta-se na metodologia de pesquisa Pesquisa-Aplicação, adequada ao desenvolvimento de intervenções como soluções pedagógicas para problemas complexos de origem educacional (PLOMP, NIEVEEN, *et al.*, 2018).

O caminho trilhado para a criação da proposta de intervenção será descrito ao longo desta monografia, que está estruturada como descrito a seguir.

O Capítulo 2, constituído pela fundamentação teórica do trabalho, discute os conceitos atrelados ao pensamento computacional, apresentando um panorama internacional de sua aplicação pedagógica, assim como, as abordagens brasileiras adotadas para o seu uso em sala de aula na Educação Básica. As ferramentas de aprendizagem vigentes atreladas ao desenvolvimento de habilidades relacionadas ao pensamento computacional também são discutidas em conjunto com questionamentos gerados a partir da revisão da literatura.

A metodologia de pesquisa Pesquisa-Aplicação, assim como os conceitos que a constituem, é introduzida e evidenciada ao longo do Capítulo 3, tendo como referência o processo descrito por Reeves (2006), em que a pesquisa acontece ao longo de 4 (quatro) fases potencialmente cíclicas que contemplam: a fundação do arcabouço teórico; o desenvolvimento

de proposta de intervenção; a implementação e validação iterativa dessa proposta e, por fim, a discussão das conclusões a partir dos princípios de design resultantes.

A proposta de intervenção e a implementação do ambiente #INCLUDE são detalhadas no Capítulo 4, ressaltando o respeito ao enfoque metodológico associado a cada etapa do processo, assim como as decisões de projeto e tecnologias envolvidas na sua concepção e implementação.

O Capítulo 5 reúne a discussão dos resultados da pesquisa a partir de considerações que nortearam a evolução do desenvolvimento do trabalho e que definiram o seu formato final até este ponto de sua apresentação. Nesse capítulo também são discutidas as propostas de extensão do trabalho resultantes da experiência e maturidade obtidas ao longo do seu desenvolvimento, apoiadas na metodologia de pesquisa adotada.

2 O PENSAMENTO COMPUTACIONAL E SUAS ABORDAGENS NA ATUALIDADE

Em 1843 a condessa de Lovelace publicou um artigo sobre a Máquina Analítica de Charles Babbage, projeto de um aparato de cálculos gerais que podia desempenhar uma variedade de operações aritméticas com base em instruções que lhe fossem fornecidas em uma espécie precursora de modelo de programação (ISAACSON, 2014). O artigo conhecido como “Notas da Tradutora” dissemina conceitos que revolucionaram a história da computação, um século antes do primeiro computador ser de fato desenvolvido. Dentre as destacadas contribuições da primeira programadora Ada Lovelace, três conceitos de programação se tornaram imprescindíveis até os dias de hoje: a definição de função ou procedimento, laço iterativo e a aplicação da lógica condicional. Jeannette Wing (2006), em seu artigo seminal, emprega também esses três conceitos para discutir o que ela denomina Pensamento Computacional, habilidade considerada para a autora tão fundamental nos dias de hoje às pessoas quanto a escrita e a matemática.

Pensamento computacional é pensar recursivamente. É processamento paralelo. (...) Pensamento computacional é usar abstração e decomposição ao enfrentar uma atividade vasta e complexa ou projetar um sistema grande e complexo. É a separação de interesses. É escolher uma representação apropriada para um problema ou modelagem dos aspectos relevantes de um problema para torná-lo tratável (WING, 2006, tradução nossa).

Os aportes que o pensamento computacional apresenta para diferentes domínios são os mesmos, independentemente do tipo de prática e do seu nível de complexidade, para o sujeito que se propôs a realizá-la. Seu embasamento deriva de processos e soluções inerentes à ciência da computação, sendo eles: a profunda compreensão do problema a ser resolvido na execução de uma atividade; a percepção das interfaces no contexto em que esse ocorre, aliado às reflexões pertinentes à sua existência; o processo mental ocasionado a partir da adoção de um caminho de solução – o qual pode até não ser o mais eficiente a princípio – que proporciona a elaboração de uma resolução completa do problema a partir da identificação das partes relevantes que, juntas, compõem a atividade como um todo; a estruturação das ações ou caminhos sequenciais de solução para cada uma das partes relevantes identificadas na etapa anterior; a execução paralela, quando couber, (pelo uso complementar de recursos de natureza tecnológica ou humana) para as etapas mais exaustivas da atividade; validação dos resultados alcançados com

o modelo acima e, eventualmente, ajustes e refinamentos relacionados com a eficiência da solução, quando necessário.

O próprio método de desenvolvimento deste texto pode ser usado para exemplificar a aplicação do pensamento computacional. Antes mesmo de escrever as suas primeiras linhas foi necessário compreender claramente o objetivo de uma monografia resultante de um Trabalho de Conclusão de Curso, o contexto de sua aplicação e o que se esperava de seu conteúdo ao final do processo de construção. Em seguida, os elementos textuais associados aos seus componentes – introdução, pesquisa bibliográfica, metodologia de pesquisa adequada, processo de desenvolvimento da solução etc. – foram identificados e assimilados antes de serem implementados. Somente após esse processo, efetuou-se escrita do texto propriamente respeitando os limites semânticos de cada uma de suas seções para, enfim, na leitura final garantir que todos os aspectos esperados de uma monografia tenham sido cobertos no produto resultante.

A partir do explanado é possível inferir que, apesar de embasar-se em conceitos seminais da computação, o pensamento computacional não se restringe à utilização deles em seu âmbito de origem. O pensamento computacional contribui em diversas áreas desde as ciências sociais, auxiliando na modelagem de reações pandêmicas, até a biologia, destacando-se no sequenciamento genômico (BUITRAGO FLÓREZ, CASALLAS, *et al.*, 2017). De fato, os três conceitos de programação delineados por Ada Lovelace apresentados no início deste capítulo foram formulados a partir da codificação de um algoritmo para computar números de Bernoulli, demonstrando assim a inerente vocação da Computação como ciência aplicada a outras áreas do conhecimento.

2.1 INICIATIVAS MUNDIAIS RELATIVAS AO ENSINO DO PENSAMENTO COMPUTACIONAL

A partir desta compreensão do pensamento computacional como habilidade interligada a diversas áreas do conhecimento e atuação humana, estabelece-se a relevância da sua difusão em um segmento maior do que apenas aquele direcionado a pessoas que estudam e se formam em computação em um curso de graduação. Países europeus já investem no ensino de computação como matéria obrigatória da Educação Básica, enfatizando a programação nessa etapa como um possível meio de desenvolver o pensamento computacional desde a infância e

juventude; a Inglaterra, por exemplo, implantou os estudos de Ciência da Computação, Tecnologia da Informação e Letramento Digital na Educação Básica contrapondo seu antigo programa de informática que antes era direcionado apenas ao ensino e utilização de softwares de escritório (VALENTE, 2016).

No contexto nacional, observa-se que os currículos escolares, de uma forma geral, ainda não abrangem o ensino do pensamento computacional como um componente curricular regular. Não obstante, a recentemente promulgada Base Nacional Comum Curricular – BNCC propõe associá-lo de forma coesa ao currículo de matemática ao longo das duas últimas etapas da Educação Básica (BNCC, 2017). Para o Ensino Fundamental, por exemplo, é destacada a relação entre a Álgebra e o pensamento computacional, tendo como pontos de congruência os conceitos de variáveis, algoritmos e identificação de padrões. Por outro lado, o documento ressalta também, numa relação inversa, como o próprio ensino de Álgebra, entre outros conteúdos presentes na matemática, estimula o desenvolvimento da habilidade do pensamento computacional.

Outro aspecto a ser considerado é que a aprendizagem de Álgebra, como também aquelas relacionadas a Números, Geometria e Probabilidade e estatística, podem contribuir para o desenvolvimento do pensamento computacional dos alunos, tendo em vista que eles precisam ser capazes de traduzir uma situação dada em outras linguagens, como transformar situações-problema, apresentadas em língua materna, em fórmulas, tabelas e gráficos e vice-versa (BNCC, 2017).

Para o Ensino Médio, a BNCC considera importante que o pensamento computacional prossiga estimulado pela Matemática, porém levando em conta questões sociais presentes nas vivências dos estudantes, como os avanços da tecnologia e os requisitos do mercado de trabalho.

Em outra frente, a Sociedade Brasileira de Computação – SBC reconhece e estimula projetos da Computação aplicáveis na Educação Básica de forma ampla, além de ter defendido o pensamento computacional em discussões com representantes da BNCC a fim de reforçar a sua presença nessas diretrizes (MARTINELLI, ZAINA e SAKATA, 2018).

Nesse cenário, um grande empecilho – talvez o maior – no ensino do pensamento computacional deriva do fato dele não ter sido difundido como formação entre aqueles que não tiveram formação na área da Computação. Como um professor da Educação Básica pode ensinar a seus alunos algo que ele nunca aprendeu ou não se sente seguro para tanto? Por outro

lado, esperar que uma nova e ampla geração de educadores se gradue trazendo consigo, desde a Universidade, o conhecimento para desenvolver o pensamento computacional nos estudantes levará anos e, mesmo assim, esta é uma expectativa profundamente dependente de decisões políticas e de alterações e ajustes em currículos de cursos de licenciaturas. Considerando esse aspecto, as ferramentas *online* apresentam-se como uma alternativa interessante para a orientação dos professores na prática do pensamento computacional com seus alunos¹.

2.2 FERRAMENTAS DE SOFTWARE COM FOCO NO ENSINO DO PENSAMENTO COMPUTACIONAL

Neste trabalho consideramos *ferramenta* qualquer recurso de software, disponível *online* ou *offline*, desenvolvido com o objetivo de demonstrar, ilustrar, exercitar, compreender e fixar conceitos abstratos de disciplinas, sem o qual o processo de ensino e aprendizagem desses conceitos fica restrito a um modelo, na maioria das vezes, unidirecional, linear e textual de transmissão de conhecimentos entre quem ensina e quem aprende. Entendemos, portanto, que as ferramentas podem ofertar formas dinâmicas, participativas, lúdicas e orientadas de construir intervenções efetivas na educação que envolvam os estudantes ativamente nesse processo.

No universo das ferramentas que contemplam o ensino do pensamento computacional na Educação Básica, no domínio das linguagens de programação baseadas em blocos², VALENTE (2016) cita três que aparecem de forma recorrente na literatura científica: Scratch³, Snap!⁴ e Alice⁵. Contudo, a proposta dessas três aplicações volta-se a um ambiente livre para a criação, de modo que seus usuários possam liberar a imaginação durante a programação de narrativas, sem necessariamente atrelar-se a um enfoque metodológico de ensino do

¹ Outras alternativas, por exemplo, incluem a robótica pedagógica e as atividades desplugadas (jogos e competições) para mostrar aos alunos formas de solução de problemas usando o pensamento computacional (VALENTE, 2016).

² Conjunto de linguagens visuais de programação que se utilizam de blocos pré-existentes encaixáveis para orientar a implementação de algoritmos, liberando os seus usuários de preocupações com a sintaxe da linguagem e permitindo que ele concentre-se nos conceitos e aspectos semânticos da programação (variáveis e constantes literais, encadeamento lógico dos comandos, repetições, etc.)

³ O *Scratch* é um projeto do Lifelong Kindergarten Group do MIT Media Lab. Disponível em: <https://scratch.mit.edu>

⁴ O *Snap!* É uma plataforma desenvolvida pela Universidade da Califórnia-Berkeley. Disponível em: <https://snap.berkeley.edu>

⁵ Alice é um ambiente de programação mantido pela Universidade Carnegie-Mellon. Disponível em: <https://www.alice.org>

pensamento computacional. O Scratch, por exemplo, dispõe de recursos pré-programados (personagens editáveis e recursos áudio-visuais) que permitem a criação de pequenos jogos e histórias interativas. Pesquisas na área frequentemente sugerem a utilização de ferramentas como as citadas aliadas à alguma metodologia potencializadora para aplicações efetivas em educação. Porém, como Hickmott, Prieto-Rodriguez e Holmes (2017) afirmam, muitos desses estudos são feitos por profissionais da Computação, sem um apoio multidisciplinar de pesquisadores da área da Educação, ocasionando assim, um hiato a ser preenchido na pesquisa do pensamento computacional.

A carência de uma orientação metodológica, com foco no desenvolvimento da habilidade do pensamento computacional, intrínseca às ferramentas supracitadas, cria mais um empecilho para a introdução do tema nas Escolas. Senin e Nasri (2019) evidenciam, a partir de uma ampla pesquisa feita com professores, que existe apreensão da categoria relativa ao ensino do pensamento computacional, entretanto também há interesse em abordar o assunto no exercício da profissão, mas julgam necessário mais informações e formação sobre como aplicá-lo na docência.

Especificamente, o Code.org⁶ apresenta-se como possível solução para este último aspecto assinalado. A organização surgiu no contexto de fomentar o ensino de ciência da computação com projetos que associam o pensamento computacional a jogos que envolvem programação e atividades desplugadas, provendo apoio pedagógico aos professores a partir de planos de aula, treinamentos e cursos que podem ser desenvolvidos pelo educador em conjunto com a turma. Em teoria, a iniciativa diminuiria a distância percebida entre os professores e o ensino do pensamento computacional, por estabelecer uma ligação entre metodologias que desenvolvem o pensamento computacional e uma ferramenta de fácil acesso e utilização. Entretanto, na avaliação de Israel e Pearson (2015) outra variável participa nessa equação: para integrar as atividades propostas pelo Code.org na sala de aula, professores pesquisados por esses autores, em sua maioria, tiveram um trabalho extra (fora da sala de aula) para adequar os conceitos do pensamento computacional com as práticas de suas respectivas matérias. Isso deve-se ao tempo restrito que é reservado ao professor para cada unidade de aula. Esse encaminhamento é reforçado também em outras referências como conduta necessária para auxiliar os alunos na aplicação do pensamento computacional em outros domínios.

⁶ Code.org é uma organização sem fins lucrativos dedicada à expansão do acesso à ciência da computação pela via principal da programação de computadores. Acessível a partir do endereço <https://code.org>

Como conclusão de diversos desses estudos, inicialmente, sugere-se que a Matemática seja a matéria escolhida para abordar o ensino do pensamento computacional, não apenas pelo fato das duas áreas de conhecimento estarem associadas pela BNCC conforme visto, como também pelo intrínseco relacionamento entre a educação matemática e o raciocínio lógico ao longo do Ensino Básico, de acordo com as conclusões de Morais, Basso e Fagundes (2017). Após estudos na área, os autores, também observaram que:

Há uma oportunidade de abordar os conceitos de matemática de novas formas, no qual o conteúdo será contextualizado e significativo para as crianças. Assim, faz-se necessário o desenvolvimento de novas experiências em sala de aula, de metodologias de ensino de matemática para que mais educadores se aventurem a ensinar matemática partir da programação (MORAIS, BASSO e FAGUNDES, 2017).

Como visto, se por um lado o pensamento computacional apresenta um grande potencial de formação para a atuação em áreas diversas do conhecimento, não se restringindo à Computação – como na Educação, por exemplo – por outro existe uma deficiência de preparo de profissionais nessas outras áreas – no exemplo citado, os professores – para explorar com efetividade o pensamento computacional na sua prática pedagógica. Uma das abordagens para tentar reduzir essa lacuna tem empregado ferramentas de programação visual baseadas em blocos, que dão liberdade para criar modelos pedagógicos associados. Contudo, mesmo nesses casos o problema retorna, já que esse ato criativo exige formação especializada. Quando as duas questões são associadas, percebemos um terceiro problema: a necessidade de integração a uma matéria da base curricular em vigência no país de aplicação.

As soluções que apresentam componentes pedagógicos – em algum nível – fornecidos pelos seus desenvolvedores são pautadas no ensino e aprendizagem de programação – como no caso dos citados ambientes Scratch, Snap! e Alice. Um efeito colateral possível nessa abordagem, corresponde à assimilação de alguns conceitos introdutórios do pensamento computacional, mesmo sem embasamento em uma metodologia que aborde essa forma de pensar especificamente. No entanto, esse fato não indica que o pensamento computacional esteja sendo tratado adequadamente. Em outro aspecto, plataformas que oferecem um programa de ensino com maior ênfase na oferta de modelos pedagógicos – como no caso do Code.org –, acabam por ser generalistas, sem envolvimento com questões socioeconômicas e culturais do país, que igualmente devem ser consideradas, posto que apresentam um impacto direto no processo de aprendizagem (BUITRAGO FLÓREZ, CASALLAS, *et al.*, 2017). Nesse sentido, as ferramentas disponíveis não se adequam em um contexto nacional no qual o pensamento

computacional configura-se apenas como uma proposta extra currículo básico, evidenciando, portanto, a inevitabilidade de sua integração a uma matéria já prevista. Nesse sentido, há carência de pesquisa e desenvolvimento de ferramentas que conduzam o professor nessa jornada e que, no processo, ele próprio desenvolva competências para a prática do pensamento computacional com seus alunos.

2.3 TRABALHOS CORRELATOS

Distintas propostas de intervenção têm sido apresentadas para suprir a lacuna de ensino do pensamento computacional na Educação Básica. Apesar de muitas delas empregarem as já citadas ferramentas Scratch, Alice, Snap e Code.org, ainda há espaço para inovação.

O aplicativo T-mind (PESSOA, ARAÚJO, *et al.*, 2017) é um bom exemplo de intervenção. Trata-se de uma ferramenta desenvolvida com o objetivo de incentivar a assimilação de habilidades relacionadas ao pensamento computacional em alunos da Educação Básica. O projeto aplica questionários com foco na resolução de problemas de forma *gamificada* visando um engajamento maior dos usuários e promovendo o estímulo de habilidades como abstração, decomposição e análise de dados.

O Caça ao Tesouro (PINHO, WEISSHAHN, *et al.*, 2016) retrata outra proposta de ferramenta com foco no ensino com pensamento computacional, neste caso um aplicativo com um jogo de tabuleiro em que o estudante assume o papel de um pirata e deve procurar um tesouro escondido por outro colega. Nesse contexto a ferramenta propõe-se a desenvolver habilidades tais como busca e abstração de dados.

Em ambos os casos existe pouca flexibilidade para a inserção de novos conteúdos por parte dos professores, tornando essas soluções pouco atrativas em propostas ampliadas de ensino do pensamento computacional.

3 OBJETO DE ESTUDO E METODOLOGIA DE PESQUISA

Como visto no capítulo anterior, existe uma lacuna de propostas que estudem o desenvolvimento de ferramentas de apoio com base tecnológica que promovam o ensino do pensamento computacional, explorando às especificidades das matérias de formação no Ensino Básico e que orientem o professor nesse processo, resultando que ele próprio desenvolva essa competência a partir de uma atuação ativa.

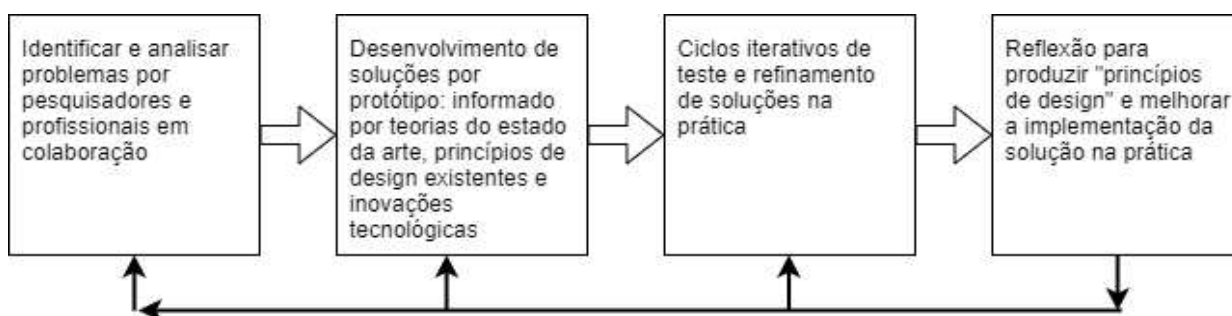
A partir desse objeto de estudo, a metodologia de pesquisa escolhida e prevalente para este trabalho fundamenta-se na abordagem conhecida como Pesquisa-aplicação, denominada também como *Design-Based Research* – DBR. Idealizada no âmbito das ciências exatas, especificamente nas áreas da Computação e da Engenharia de Produção, ela similarmente encontrou destaque como um dos processos metodológicos de pesquisa válidos para intervenções na área da Educação. Plomp, Nieveen, *et al.* (2018) a definem como:

Uma abordagem apropriada para desenvolver soluções a partir de pesquisa científica para problemas complexos no contexto da prática educacional, ou ainda desenvolver ou validar teorias sobre processos de aprendizagem, ambientes de aprendizagem e assemelhados.

Uma das principais características da Pesquisa-aplicação consiste no duplo produto referido em seu próprio nome, tanto a pesquisa quanto a intervenção fundamentada nesta. O componente de pesquisa manifesta-se, independente do resultado gerado a partir da intervenção desenvolvida, como contribuição para futuros estudos. Esse fator dialoga diretamente com a área da Computação, uma vez que, a produção científica nesse campo tende a contemplar essas duas esferas.

O processo da Pesquisa-aplicação aparece em diversas fontes literárias ilustrado de formas distintas. Reeves (2006) sintetizou tal processo em um método que contempla quatro fases cíclicas, sendo essas: análise e exploração do problema; desenvolvimento de soluções utilizando princípios de design e inovações tecnológicas; implementação e avaliação em ciclos iterativos e, por fim, produção de princípios de design próprios. A relação entre essas fases está ilustrada na Figura 1.

Figura 1 – Refinamento de problemas, soluções, métodos e princípios de elaboração



Fonte: Reeves (2006)

A primeira fase, análise e exploração do problema, refere-se a uma revisão da literatura em torno do objeto de estudo. Não obstante, essa fase concerne também às questões cotidianas daqueles que estão envolvidos no ambiente ao qual o estudo será concebido. Com base no arcabouço teórico constituído nessa etapa inicial de pesquisa, busca-se a definição dos princípios de design, que implicarão no projeto do protótipo da solução e que futuramente serão revisados nas etapas seguintes da metodologia.

Princípios de design são afirmações que podem ser inferidas a partir de trabalhos acadêmicos em relação ao objeto de estudo. Como descrito por McKenney e Reeves (2012), eles agem como recomendações para o tratamento de uma classe específica de problemas com diferentes configurações (*apud* EULER, 2017). A título de exemplo, neste trabalho um princípio de design definido é o ensino inicial do pensamento computacional por parte de professores de matemática na Educação Básica, inspirado nas propostas de Moraes, Basso e Fagundes (2017);

A segunda fase trata da concepção e desenvolvimento da solução, tendo como referência reflexões pertinentes ao estado da arte e aos princípios de design que irão alicerçar a estruturação da aplicação. Ao final dessa etapa, espera-se obter uma proposta de solução robusta para o problema da pesquisa.

Após a concepção da solução, inicia-se a terceira fase, implementação e avaliação. Nessa etapa do processo, como o nome sugere, a aplicação será desenvolvida de forma concreta e submetida a uma avaliação qualitativa, por especialistas na área ou grupos focais. Subsequente

a primeira avaliação, a solução será analisada e redesenhada para atender as questões levantadas durante a avaliação e então passará pelo processo novamente. Esta etapa possui seu próprio aspecto iterativo interior ao ciclo de fases da Pesquisa-aplicação e é, preferencialmente, concluída quando um equilíbrio entre o ideal e o realizado é atingido (PLOMP, NIEVEEN, *et al.*, 2018).

A quarta fase, produção de princípios de design, diz respeito ao conhecimento adquirido ao longo do processo e como ele se traduz em princípios que poderão futuramente ser aplicados por diferentes pesquisadores. A síntese da solução em princípios de design pode ou não ser refinada e posteriormente publicada, assim como pode dar início a um novo ciclo da Pesquisa-aplicação (HERRINGTON e REEVES, 2011).

Neste trabalho, a primeira etapa da Pesquisa-aplicação – revisão da literatura – alinha-se com dois de seus objetivos específicos – identificação de um conteúdo de formação da Educação Básica que propicie uma abordagem efetiva para formação em/a partir do pensamento computacional e definição dos princípios de design que devem ser contemplados pela solução para nortear a construção de um protótipo – uma vez que, ambos necessitam de um forte embasamento teórico tecido a partir da revisão literária.

O desenvolvimento de uma proposta de intervenção com base nos princípios de design resultantes do processo metodológico aplicado, que constitui mais um dos objetivos específicos deste trabalho, reflete-se na segunda fase da Pesquisa-aplicação. Sua realização é contemplada pelo processo iterativo de desenvolvimento e avaliação, que tem como propósito uma solução refinada para o problema de pesquisa proposto. A terceira fase da Pesquisa-aplicação é aderente à avaliação proposta no quarto objetivo específico deste trabalho, que corresponde a definição da comunidade de validação para o ambiente de aprendizagem proposto e avaliar a solução em um contexto que envolva componentes dessa comunidade da Educação Básica em ciclos que promovem a melhoria da proposta de intervenção.

As métricas definidas para nortear a avaliação da intervenção são delineadas pela própria metodologia. A Pesquisa-aplicação almeja intervenções de alta qualidade (PLOMP, NIEVEEN, *et al.*, 2018). Para atingir esse objetivo quatro critérios devem ser adotados: *relevância*, diz respeito a validade do conteúdo, pautada na revisão sistemática da literatura no tema; *consistência*, refere-se à validade da construção do protótipo, a intervenção deve ser projetada com coerência; *praticidade*, concerne ao entendimento de que a intervenção faz

sentido para o contexto a qual foi desenvolvida; e por fim, *efetividade*, que compete aos resultados esperados ou reais acerca da intervenção, se foram ou possivelmente serão atingidos. A elaboração efetiva dos instrumentos de avaliação do projeto é constituída ao longo do processo e deve respeitar a metodologia, seguindo os critérios elencados.

Ao final desse processo, espera-se deter uma solução pautada a partir de uma metodologia de investigação que confere a tal solução um aporte de base metodológico-conceitual e fundamenta sua robustez, suprimindo assim o objetivo geral da pesquisa.

4 DESENVOLVIMENTO DA SOLUÇÃO GUIADO PELA PESQUISA-APLICAÇÃO

Neste capítulo a proposta de intervenção realizada será descrita em seções que representam cada fase preconizada pela Pesquisa-aplicação detalhada no capítulo anterior. Optou-se por esta abordagem em função da proximidade inequívoca entre o percurso do enfoque metodológico proposto e o desenvolvimento progressivo da solução. A primeira seção apresenta a descrição do processo inicial da pesquisa; a segunda seção trata dos princípios de design considerados ao propor a solução; a terceira seção descreve o processo iterativo de implementação e avaliação do protótipo de ambiente como proposta de intervenção e, por fim, a quarta seção trata dos princípios de design inferidos a partir deste processo.

4.1 ANÁLISE E EXPLORAÇÃO DO PROBLEMA

Uma revisão sistemática em torno do tema foi realizada com a finalidade de evidenciar caminhos que pudessem esclarecer não apenas como o pensamento computacional tem suscitado pesquisas envolvendo sua aplicação na Educação Básica, como levantar ferramentas e metodologias que fundamentam práticas pedagógicas, adotadas por professores e pesquisadores, envolvendo o ensino e aprendizagem dessa forma de pensar em níveis diversos na Educação Básica. Além de conhecer também os resultados alcançados por essas pesquisas até o momento no contexto em que elas foram constituídas, esse processo subsidiou o referencial teórico descrito no Capítulo 2 deste trabalho.

4.2 DESENVOLVIMENTO DA SOLUÇÃO UTILIZANDO PRINCÍPIOS DE DESIGN

A segunda fase teve o intuito de propor uma intervenção pautada no estudo da temática realizado na etapa anterior da pesquisa. Baseado nele, foi então possível construir dois princípios de design fundamentais para a proposta de intervenção:

1. A educação matemática e o raciocínio lógico estão intrinsecamente associados durante a Educação Básica, portanto esta área do conhecimento é adequada para

o ensino introdutório do pensamento computacional. Esse princípio pode também ser inferido a partir dos estudos de Morais, Basso e Fagundes (2017).

2. A programação em blocos visuais melhora o aprendizado de conceitos relacionados ao pensamento computacional sem a carga cognitiva de dominar uma linguagem de programação textual (SAEZ-LOPEZ, ROMAN-GONZALEZ e VAZQUEZ-CANO, 2016).

A partir dos princípios de design elencados acima foi possível desenvolver a ideia inicial da proposta de intervenção.

4.3 IMPLEMENTAÇÃO E AVALIAÇÃO EM CICLOS ITERATIVOS

Com base nos princípios de design definidos e fundamentados na análise do estado da arte, trilhou-se o caminho de constituição da terceira fase preconizada pela Pesquisa-aplicação que culmina na modelagem de um protótipo como plano de intervenção a ser avaliado e refinado.

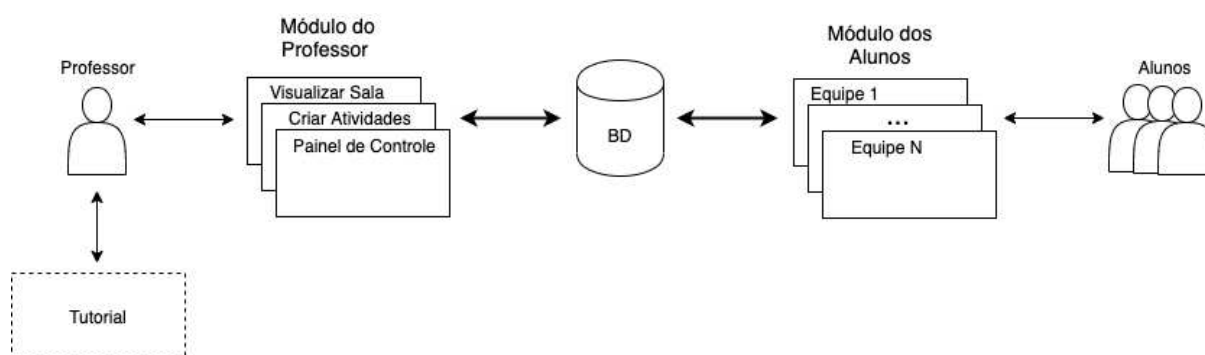
Coerentemente com Pressman e Maxim (2015), que definem princípios de Engenharia de Software e são referência nesta área, quando existem objetivos delineados para sistemas de software sem o detalhamento dos requisitos essenciais para alcançá-los, o modelo de processo conhecido como prototipação é uma possibilidade viável e prevalente como fase preliminar ao desenvolvimento do software. Esse processo consiste em 4 (quatro) etapas cíclicas que iniciam a partir da comunicação entre as partes envolvidas e evoluem para um planejamento, seguido por uma modelagem da ferramenta, criação de um protótipo e, por fim, a apresentação do que foi desenvolvido ao usuário, para assim compreender, contribuir, elaborar e avaliar os requisitos que devem ser atendidos. Alinhado com esse preceito da Engenharia de Software, neste ponto do trabalho foi elaborado um protótipo não operacional com o objetivo de servir como prova de conceito visual da dinâmica prevista de funcionamento da solução tendo em vista facilitar a compreensão e avaliação do especialista ao longo do primeiro ciclo avaliativo.

A concepção do protótipo da solução considerou em todas as suas etapas a abstração das atividades regularmente desenvolvidas por professores (os principais usuários), sempre que uma decisão de projeto se mostrou necessária nessa primeira fase. Como exemplos, planejar atividades pedagógicas para uma aula, conduzir a turma para uma sala de atividades específicas

e controlar o desenvolvimento das atividades por seus alunos. No protótipo, conceitos como sala e atividade emergiram dessa perspectiva tendo em vista facilitar a aceitação inicial dos professores, já que ele encontrará no ambiente um espaço amigável e coerente com a sua rotina de trabalho.

A visão geral dos principais componentes tecnológicos do protótipo, os relacionamentos entre eles e como os usuários (professores e alunos) interagem com o ambiente está esboçado na Figura 2 abaixo. Para compreensão neste ponto do trabalho detalharemos as telas dos módulos do professor e dos alunos. Os demais componentes serão esclarecidos ao longo do texto.

Figura 2 – Visão geral da solução

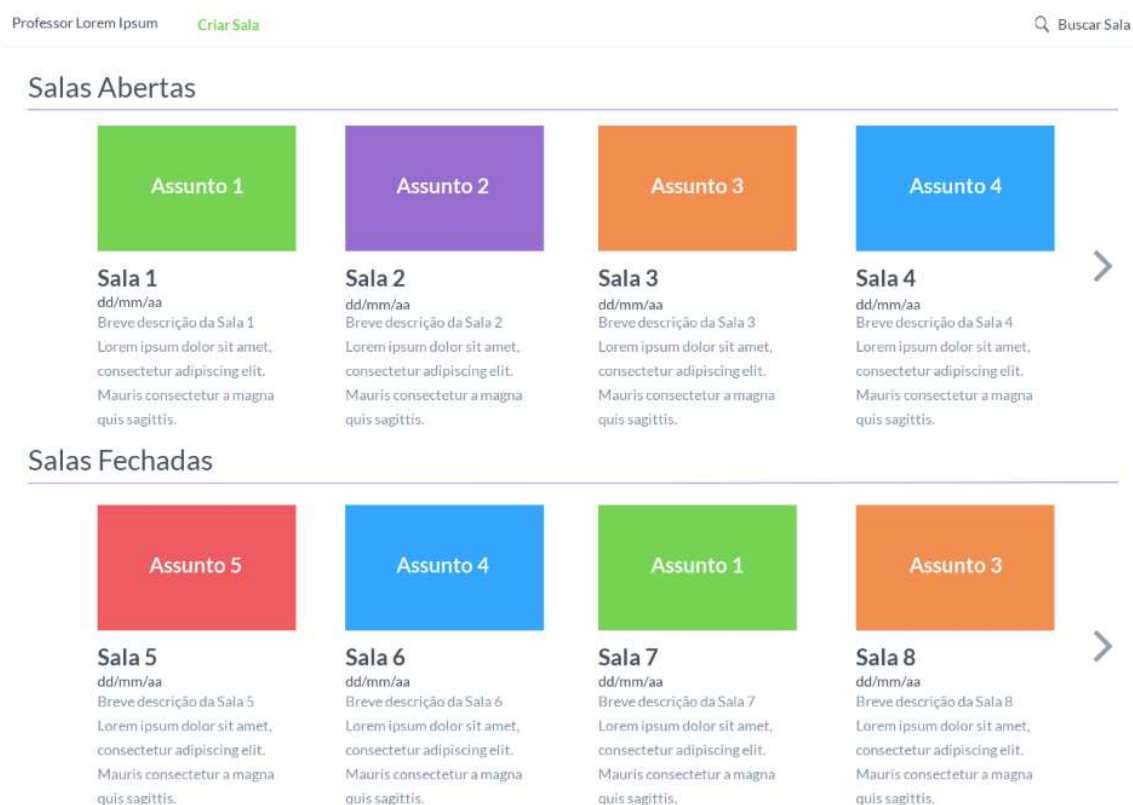


Fonte – A autora

4.3.1 Descrição do Protótipo

Nesta seção serão descritas as principais características do protótipo e o inter-relacionamento pretendido dos usuários (professores e alunos) com os módulos da ferramenta. É importante ressaltar que algumas das funcionalidades descritas foram modificadas como resultados naturais do enfoque metodológico da Pesquisa-aplicação (ciclos avaliativos com especialistas). Uma sala é criada pelo professor-usuário do ambiente para realizar uma atividade, que será escolhida pelo mesmo a partir de um banco de dados que lhe apresenta exercícios classificados por assuntos definidos, por exemplo, na BNCC. Como apresentado na Figura 3, a abstração *sala* pode assumir dois estados, *aberta* e *fechada*; o primeiro deles corresponde a uma sala com seu exercício já definido, porém não realizado, e o segundo, um histórico de exercícios já aplicados pelo professor em suas respectivas turmas para referências futuras.

Figura 3 – Tela inicial do Professor



Fonte – A autora

Após criar uma sala o professor poderá registrar uma atividade que será realizada em sua aula. Esse processo inicia-se a partir da definição do assunto que será trabalhado e então segue para a escolha do exercício – a partir de opções previamente formatadas e armazenadas em uma base de dados da ferramenta, elaboradas para suprir tanto os aspectos pedagógicos do assunto trabalhado quanto os do pensamento computacional a ele associado. Por fim, usando para efeito de exemplo o assunto equações de primeiro grau, a Figura 4 apresenta as definições dos dados de entrada como valores dos coeficientes **a** e **b** na equação, e saídas esperadas – os resultados que os alunos devem encontrar ao resolvê-las. É importante ressaltar que o professor também poderá inserir novas entradas e saídas esperadas para o problema selecionado, participando ativamente do processo de configuração da atividade.

Figura 4 – Tela referente a criação da atividade

Fonte – A autora

Ao final do processo de elaboração da atividade, a sala será criada e um código de acesso será disponibilizado ao professor, que possibilitará, posteriormente a conexão dos alunos com essa atividade. No momento da sua realização, o professor informa o código da mesma aos alunos para que eles tenham acesso à atividade e possam solucionar os exercícios em suas respectivas máquinas, a partir da tela/visão do estudante. Esse código estará acessível ao professor, no momento da execução da atividade, em um ambiente do tipo Painel de Controle – conforme apresentado na Figura 5, no qual ele poderá visualizar também a descrição da atividade com suas respectivas entradas e saídas, além do estado das equipes de alunos que estão em atividade prática e seu andamento: *finalizado* e *fazendo*.

Figura 5 – Tela do Painel de Controle para o gerenciamento de uma atividade

Professor Lorem Ipsum Dashboard

Descrição da atividade:
 Lorem ipsum dolor sit amet, consectetur adipiscing elit. Mauris consectetur a magna quis sagittis. Quisque vitae viverra ax + b = 0 augue, ut pulvinar nisl. Donec nec urna auctor, mollis orci sed, consequat velit

Entradas:

a = 5; b = 10

a = y; b = y

a = z; b = z;

Saídas esperadas:

x = -2

x = k

x = w

Código: e7xbY

Equipe 1	Fazendo
Equipe 2	Finalizado
Equipe 3	Fazendo
Equipe 4	Fazendo
Equipe 5	Fazendo
Equipe 6	Fazendo
Equipe 7	Fazendo

Finalizar atividade ✓

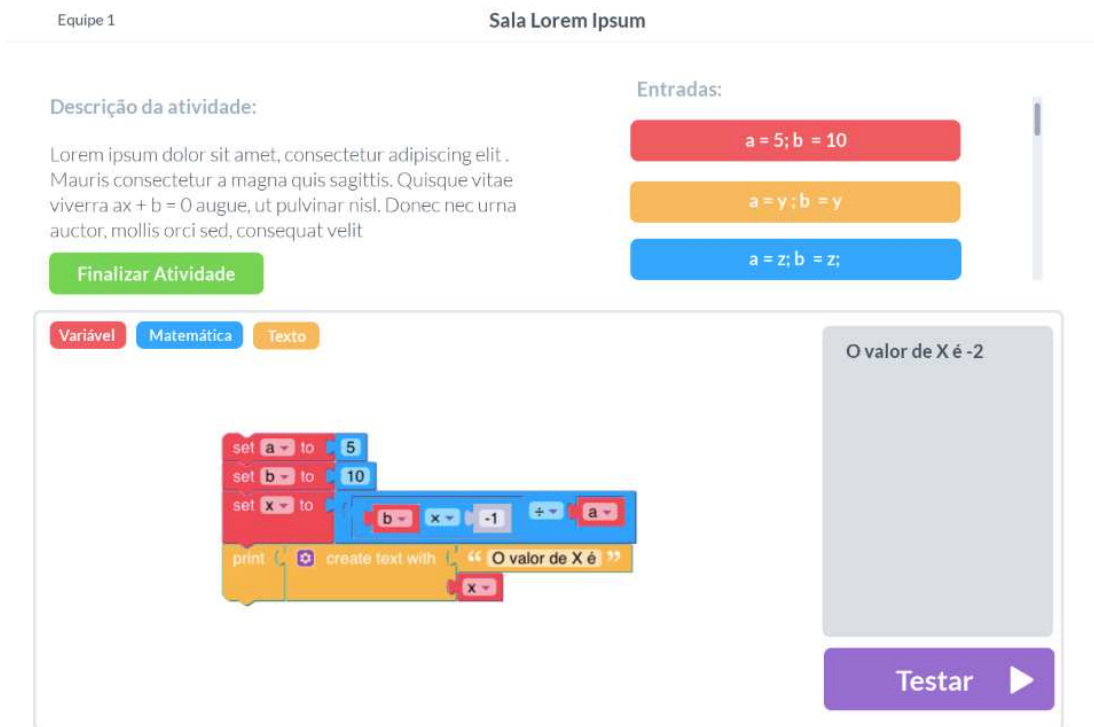
Fonte – A autora

Do ponto de vista do aluno-usuário da ferramenta, ao acessar a sala a partir do código disponibilizado pelo professor, será fornecido o enunciado da atividade assim como os valores de entrada definidos pelo professor. Seu objetivo é solucionar a tarefa usando a programação por blocos, como visto na Figura 6. Os blocos de programação são implementados usando a interface de programação de aplicações *Blockly*⁷ do Google, separados em categorias relacionadas à sua atuação no processo de solução. No exemplo da Figura 6, tais categorias são representadas pelos objetos *variáveis*, *matemática* e *texto*. O botão *testar* possibilita que o aluno possa ver os resultados da execução do seu código de forma a emular uma espécie de terminal de computador, aproximando a atividade ao cotidiano da programação de forma lúdica. Assim que o aluno desenvolver e testar seu exercício (concretização da solução do problema usando o pensamento computacional) ele poderá então clicar em *finalizar atividade*, implicando na verificação do código produzido de acordo com as entradas e saídas fornecidas previamente como parte da atividade. Caso o código de blocos esteja correto, a equipe tem o status da sua

⁷ *Blockly* é uma biblioteca JavaScript que possibilita a criação de editores para linguagens baseadas em blocos. Disponível em: developers.google.com/blockly/

atividade alterado para *completo*; caso contrário, é informado para quais entradas o código de blocos apresentou problema para que, assim, possam seguir com a depuração e correção.

Figura 6 – Tela do aluno em que será realizada resolução da atividade.



Fonte – A autora

4.3.2 Entrevista com especialista para avaliação do protótipo

De acordo com as etapas previstas na metodologia para esta pesquisa, o protótipo foi apresentado e discutido com uma especialista na área da Educação⁸, professora da UNEB, com formação em Pedagogia e em Tecnologia da Informação e titulação de Doutora em Educação e Contemporaneidade; a especialista atua ainda com projetos de pesquisa com crianças que programam com jogos digitais e o processo de multiletramento digital a partir da interação com linguagem de programação em blocos em contexto escolar. A interação com a especialista ocorreu por meio de uma entrevista presencial, na sala do Grupo de Geotecnologias, Educação e Contemporaneidades (GEOTEC), no prédio do Programa de Pós-graduação em Educação e Contemporaneidade (PPGEduC).

⁸ Esta pesquisa foi registrada e aprovada pelo Comitê de Ética em Pesquisa da UNEB sob número 26687119.8.0000.0057

No início da entrevista apresentamos a proposta do projeto, seus objetivos, a motivação e justificativa para o trabalho, assim como, o seu contexto e a metodologia de pesquisa adotada para o estudo. Foi solicitada também a permissão para a gravação do áudio da entrevista para facilitar a análise posterior desse processo avaliativo, e também, das sugestões de ajustes e melhorias conforme preconizado na Pesquisa-aplicação⁹.

Em seguida, após esclarecermos o foco do trabalho na temática do pensamento computacional e na identificação da lacuna na formação dos professores para a sua aplicação em atividades inovadoras com alunos da Educação Básica, expomos o fluxo de informações da solução usando como referencial as imagens das telas de professor e alunos constituídas na versão inicial do protótipo. Ao longo desse processo as dúvidas manifestadas pela entrevistada sobre a dinâmica de funcionamento do protótipo foram sendo elucidadas.

Posteriormente, foi cedida a palavra para a avaliação livre e geral da especialista a respeito da proposta do projeto, do encaminhamento de solução adotado, da versão inicial do ambiente e seus componentes, do modelo de desenvolvimento centrado na programação por blocos e demais considerações que ela julgasse importantes e necessárias para a evolução do projeto.

A seguir descrevemos os principais aspectos e resultados desse ciclo avaliativo inicial que promoveram a melhoria do protótipo e encaminhamento para a implementação.

O primeiro aspecto da entrevista, diz respeito à validação da proposta. A especialista ratificou a lacuna de pesquisa identificada ao longo do processo de revisão da literatura, além de corroborar a proposta do ambiente ao afirmar sobre a necessidade de se trabalhar o pensamento computacional de forma a abranger a multidisciplinariedade durante o processo de ensino-aprendizagem.

Um segundo resultado do ciclo avaliativo, consistiu na sugestão de alguns ajustes relacionados ao design do protótipo, consistindo em: alteração do posicionamento dos botões de finalizar a atividade, tanto na tela do Painel de Controle, quanto na tela do aluno; diminuição na quantidade de salas abertas e fechadas apresentadas na tela inicial do professor. Esses ajustes de design justificam-se diretamente com a percepção que professores e alunos podem ter ao fazer uso da ferramenta, com vistas a tornar a experiência mais intuitiva. Ressalto a importância

⁹ A gravação da entrevista tem o papel também de registro documental desta etapa da metodologia de pesquisa.

do fato da especialista não ter tido acesso anterior a qualquer dos aspectos indicados acima, o que demonstra um *feedback* positivo de percepção de experiência do usuário que resultou na proposta de melhoria da ferramenta.

Uma fragilidade que percebemos ao conduzir esse ciclo avaliativo relacionou-se com a ausência de dinâmica ao explicarmos o fluxo das telas, devido ao fato da apresentação ter sido realizada com impressões em papel das mesmas; a entrevistada expressou dúvidas, por exemplo, em relação ao processo de criação de atividades por parte do professor.

A etapa descrita atuou, portanto, também como prova de conceito e serviu para a incorporação de melhorias que foram novamente revisadas na fase seguinte, após a implementação do ambiente e sua apresentação em funcionamento.

4.3.3 Implementação da solução

Este ciclo iniciou-se pela definição das tecnologias que permeariam a construção do ambiente. Partiu-se da definição do seu próprio nome. Optou-se por denominá-lo #INCLUDE em alusão às diretivas de inclusão de bibliotecas de código pré-existentes em algumas linguagens de programação e, no aspecto humanista, corroborar a proposta da solução de inclusão do pensamento computacional nas atividades pedagógicas rotineiras dos professores e alunos. A seguir apresentamos, a título de registro científico, uma explanação mais técnica, porém sucinta, das decisões que nortearam a implementação da solução.

Estabelecida como uma aplicação web, o modelo cliente-servidor fundamentou o processo de implementação do #INCLUDE. O módulo cliente, em desenvolvimento web conhecido como *front-end*, relaciona-se diretamente com o usuário por meio de uma interface visual e troca de dados e informações com o servidor, nominado *back-end*, que contempla a gestão de requisições e acesso ao banco de dados.

Na concretização sob a forma de códigos de programação do modelo descrito foram utilizados *frameworks* – conjunto de funções que executam funcionalidades genéricas à maioria das aplicações, como por exemplo, gestão de rotas de navegação e conexão com banco de dados (WOHLGETHAN, 2018). Para o *front-end* a escolha do *framework* levou em conta aspectos como reusabilidade do código e a possibilidade da migração para uma plataforma *mobile* (telefone celular), enquanto que para o *back-end*, considerou-se necessário, além da

possibilidade de migração do código, um bom desempenho relativo ao processamento das requisições feitas ao banco de dados (listagem das salas, busca de atividades, registros de dados de entrada e saída das atividades, entre outras).

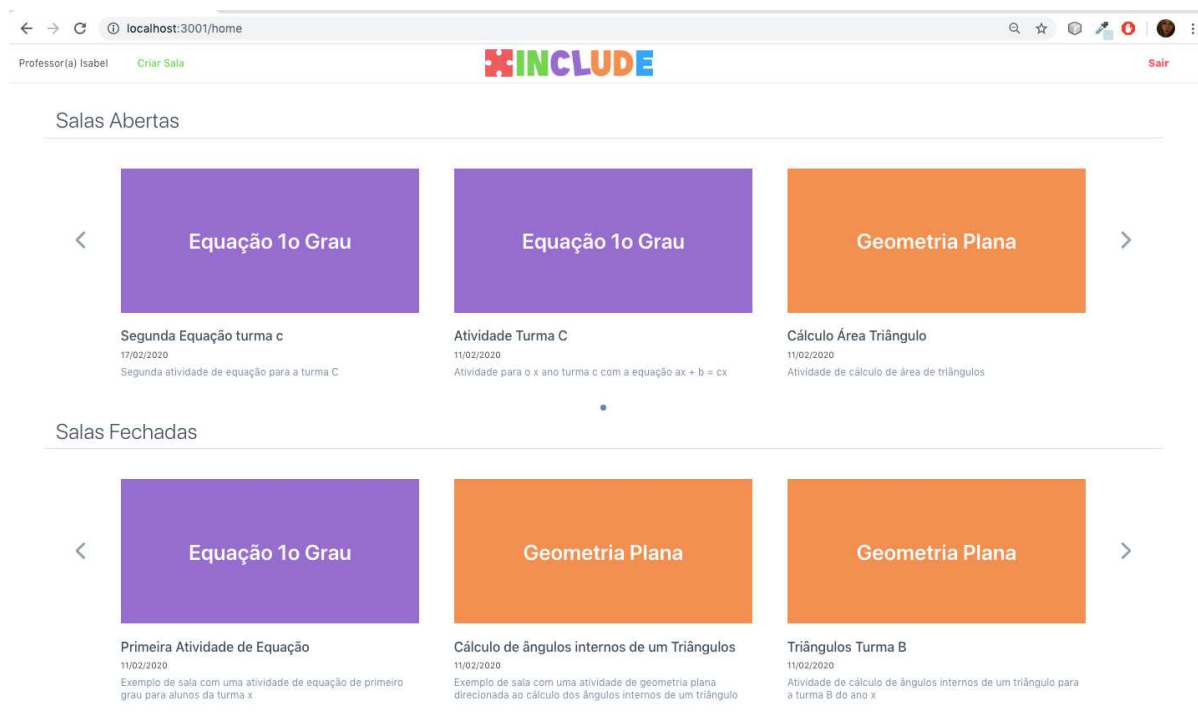
Desenvolvido em JavaScript por engenheiros da rede social Facebook, o framework React foi escolhido para a implementação do *front-end*. Baseado em componentes – pequenas partições de código independentes e reutilizáveis que permitem a composição da interface de usuário –, o React proporciona a elaboração de páginas web a partir de uma programação declarativa que pode ser transposta para aplicações móveis por intermédio da sua versão elaborada para desenvolvimento nativo, o React Native (FACEBOOK INC, 2020).

Em relação ao *back-end*, a plataforma Node.js, desenvolvida também para a linguagem JavaScript, foi escolhida visando a criação das interações com o banco de dados a partir de um servidor de programas. Sua arquitetura singular permite o desenvolvimento de sistemas escaláveis com consumo mínimo de recursos computacionais, além da possibilidade de criação de Application Programming Interfaces – API (Interfaces de Programação de Aplicativos), conjunto de funções que podem ser consumidas por outras aplicações, o que, similarmente, possibilita a migração do ambiente para o âmbito de dispositivos móveis, por exemplo (OPENJS FOUNDATION, 2020).

Por fim, a gerência da base responsável por armazenar todos os dados produzidos e manipulados durante a execução do #INCLUDE foi realizada pelo Sistema Gerenciador de Banco de Dados – SGBD MySQL que provê um serviço baseado na linguagem *Structured Query Language* – SQL orientada a bancos de dados relacionais (ORACLE CORPORATION, 2020) adotada na solução.

A ilustração da estrutura do banco de dados do #INCLUDE, por meio do seu modelo de entidade e relacionamento, é apresentado na Figura 7. Evidenciamos a tabela *sala* como elemento de destaque na modelagem, uma vez que todas as principais funcionalidades do sistema estão atreladas a ela direta ou indiretamente, nesse último caso por meio do relacionamento com tabelas intermediárias.

Figura 8 – Versão Final da Tela Inicial do Professor

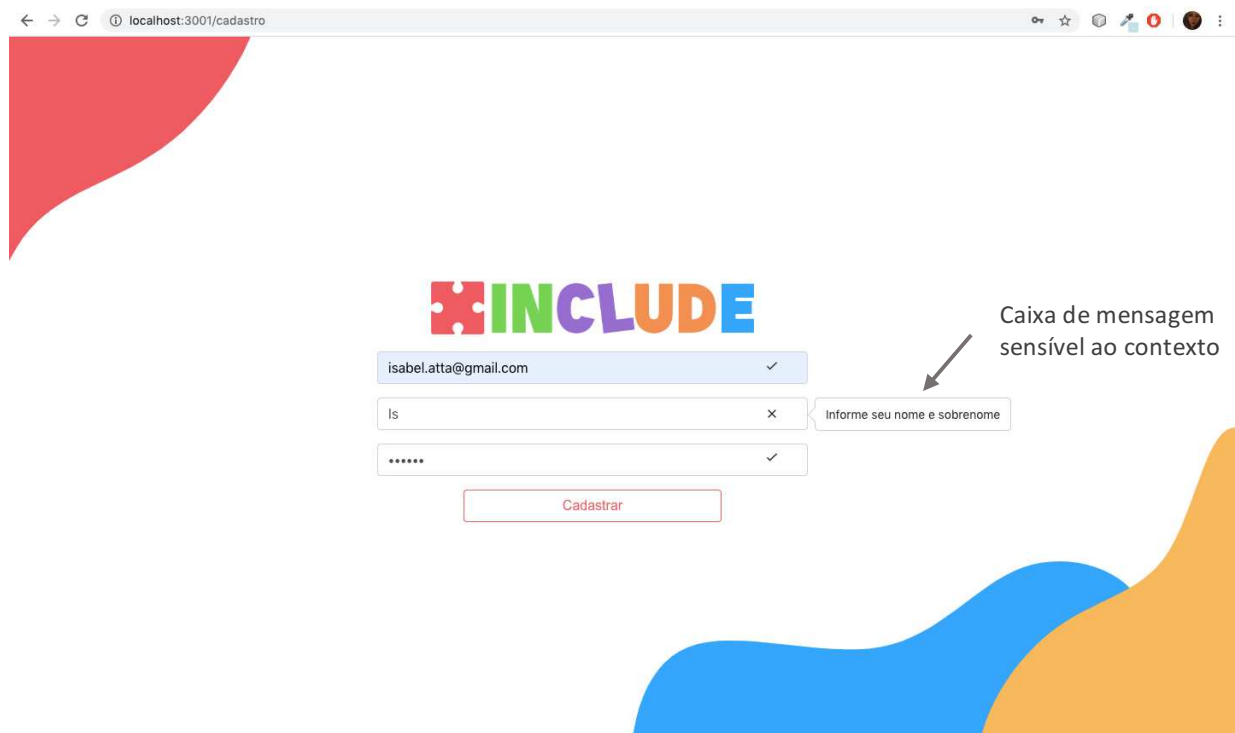


Fonte – A autora

Para contemplar as necessidades de um ambiente operacional, outras funcionalidades foram projetadas e implementadas ao longo do desenvolvimento do #INCLUDE, tais como: *Login* de professores; Cadastro de usuário; Cadastro de salas; Inserção de novas entradas e saídas para uma atividade; Cadastro de equipes. As telas correspondentes às funcionalidades citadas encontram-se na seção Apêndice deste trabalho.

Além das funcionalidades indispensáveis já mencionadas, na busca por valorizar uma solução intuitiva e amigável ao professor, foram implementadas guias no formato de caixas de texto sensíveis ao contexto que informam o que é esperado no preenchimento de campos de entrada. Funciona, portanto, como orientação no processo interativo com a solução, visualmente ilustrando e validando as entradas de informações, emitindo avisos caso algum dado tenha sido informado fora dos padrões pré-definidos para utilização do #INCLUDE. A Figura 9 demonstra esta função na tela de cadastro do professor com a indicação de uma caixa de mensagem sensível ao contexto.

Figura 9 – Tela de cadastro do professor



Fonte – A autora

Para o módulo do aluno¹⁰, na implementação da funcionalidade de programação por blocos elegemos a biblioteca Blockly, desenvolvida com o intuito de fornecer editores de códigos visuais para aplicações em JavaScript visando práticas educacionais (GOOGLE DEVELOPERS, 2020). A Figura 10 ilustra a aplicação da biblioteca no #INCLUDE usando a página de resolução da atividade de alunos como modelo.

Apesar de contemplar as funcionalidades definidas no protótipo, o módulo do aluno foi repensado durante a sua implementação para aproximar-se do cotidiano da programação e atender de uma forma mais adequada às demandas da ferramenta, considerando-se nesse processo as restrições de prazo para finalizar a implementação. Isto posto, para o #INCLUDE, a correção das atividades é realizada toda vez que o aluno testa o seu código. Esse procedimento depende diretamente de três fatores: dos valores de entradas inseridos pelo aluno no código; do cálculo correto do valor de saída por esse código; e do formato de representação desse valor de

¹⁰ Para efeito de explicação das funcionalidades do ambiente #INCLUDE usaremos o termo “aluno” para referenciar a visão que a equipe de alunos (que poderá ser constituída por um ou mais alunos) terá ao usar o #INCLUDE.

saída por meio da impressão do resultado similar ao exemplo de saída esperada indicado na descrição da atividade.

Caso o aluno realize a atividade e esqueça de apresentar o resultado da atividade usando comandos específicos para isso no próprio código, o #INCLUDE sinalizará a necessidade de inserção do bloco “imprimir” com uma mensagem de alerta associada. Atendido este aspecto, o resultado da execução pode ser apresentado ao aluno e é exibido em uma caixa de mensagem no topo da tela com os respectivos valores. Esse modelo emula a forma incipiente de apresentação de dados ensinada em cursos de algoritmos e é muito amigável aos iniciantes.

Uma vez que o aluno acerta a resposta da atividade para uma determinada configuração de dados de entrada, o #INCLUDE indica por meio de um ícone de estrela associado a entrada selecionada que ele obteve sucesso em seu cálculo – funciona como um emblema motivando o aluno para a prática com os próximos exemplo de entrada. Após a validação dos resultados associados a todas as entradas, o botão “Finalizar Atividade” fica disponível para que o aluno sinalize ao professor que terminou a atividade proposta.

Figura 10 – Módulo de atividade - visão do aluno

Isabel Segunda Equação turma c

Descrição da Atividade

Resolva a Equação $ax + b = 0$ para os determinados valores de a e b

Exemplo de Saída:

Entradas

- a = 5; b = 10 ★
- a = 10; b = 5
- a = 20; b = 60

Matemática
Variável
Texto
Lógica

```

definir a para 5
definir b para 10
definir x para b / x - -1 / + a
definir resultado para x = z
para resultado anexar texto x
imprime resultado
  
```

Módulo do #INCLUDE implementado com a biblioteca Blockly

Testar

Fonte – A autora

A partir das diretrizes mencionadas e seguindo o protótipo resultante das melhorias sugeridas no primeiro ciclo de avaliação, o #INCLUDE foi implementado e resubmetido à avaliação em um segundo momento, dessa vez, atendendo também a aspectos funcionais.

4.3.4 Avaliação de #INCLUDE – Segundo ciclo avaliativo

O segundo ciclo de avaliação do #INCLUDE objetivou validar o ambiente de acordo com os quatro critérios de avaliação explicitados pela metodologia de pesquisa adotada - *relevância, consistência, praticidade e efetividade*. Para tal, a primeira etapa contemplou a definição dos especialistas a serem entrevistados e instrumentos a serem aplicados, visando um enfoque qualitativo de análise do #INCLUDE em pleno funcionamento. A segunda etapa foi constituída pela execução propriamente da entrevista e coleta de dados – áudio dos discursos e registro escrito a partir de um questionário. Como sujeitos dessa fase da pesquisa foram eleitas duas professoras com formação e experiência ou atuação no ensino fundamental da matemática, e com elevada compreensão do contexto educacional para qual o #INCLUDE foi projetado. Para o registro do discurso e percepções das especialistas, o método de coleta de dados definido compreendeu uma entrevista semiestruturada que procedeu em quatro momentos: apresentação do #INCLUDE, demonstração de funcionamento, livre expressão oral das entrevistadas guiada pela pesquisadora sobre suas percepções a respeito do #INCLUDE e preenchimento de um questionário semiestruturado.

A entrevista completa ocorreu em uma única seção de cerca de 3 (três) horas de duração realizada na sala do Grupo de Pesquisas em Geotecnologias, Educação e Contemporaneidade – GEOTEC na UNEB. Todas as etapas de diálogo do processo foram registradas em gravação consentida de áudio para permitir uma melhor análise posterior dos discursos. O primeiro momento da entrevista compreendeu uma apresentação dos conceitos relacionados ao pensamento computacional, o objeto de estudo, a metodologia de pesquisa com o objetivo de contextualizar a proposta às entrevistadas. Em seguida, realizou-se uma demonstração do funcionamento do #INCLUDE, explicitando todas as suas funcionalidades e oportunizando um contato inicial das especialistas com a implementação da ferramenta. Logo após, foi dada a oportunidade de livre expressão oral das entrevistadas, motivada e conduzida por perguntas da pesquisadora visando captar a compreensão das mesmas sobre os princípios do #INCLUDE e percepção das potencialidades do #INCLUDE para a aplicação no contexto da Educação Básica. Por fim, procedeu-se a aplicação de um questionário escrito, focado na reflexão sobre as discussões ocorridas nos três momentos anteriores; esse instrumento específico de coleta de

dados foi concebido tendo como referência os quatro critérios de avaliação preconizados pela metodologia de pesquisa (relevância, consistência, praticidade e efetividade), compreendendo o recorte de funcionalidades principais do #INCLUDE apresentado às entrevistadas. Nesse sentido, o instrumento apresenta questões cujas respostas estão subdivididas em duas partes, sendo uma delas de respostas objetivas inspirado no modelo da escala Likert¹¹ e a outra voltada ao registro escrito livre. Uma cópia do instrumento está disponível na seção de Apêndices deste trabalho.

A seguir apresentamos a análise dos resultados dessa etapa da avaliação, organizada segundo os quatro critérios principais da metodologia, com a indicação das respostas das entrevistadas ao questionário nas respectivas questões objetivas relacionadas à cada critério e a análise do texto no caso dos registros escritos na parte aberta da entrevista.

Relevância

Ambas as entrevistadas concordaram totalmente que #INCLUDE apresenta uma abordagem diferenciada das propostas existentes para o ensino da matemática com a produção de intervenções que possuem suporte tecnológico e incentivo ao desenvolvimento da habilidade do pensamento computacional, na esfera dos seus respectivos conhecimentos e/ou que já tiveram a oportunidade de experimentar. A análise do registro escrito corrobora essa percepção ao considerar que o #INCLUDE possibilita a “*construção de conhecimentos sobre conceitos matemáticos*”, apresenta a professores e alunos “*uma interface confortável e de fácil utilização*”. Uma das entrevistadas fez o seguinte registro a respeito desse critério:

“Essa ferramenta¹² possibilita também que o aluno aprenda não só a pensar computacionalmente, mas a pensar, interpretar e verificar que existem diferentes formas de resolver um problema.”

Constatamos nesse critério que as entrevistadas destacaram também a forte vocação do #INCLUDE para o próprio ensino da matemática em si e não apenas em priorizar o ensino do pensamento computacional.

¹¹ A escala Likert foi adotada por permitir de forma simples e clara a coleta estatisticamente tratável de percepções, experiências e sentimentos dos entrevistados em relação ao objeto de pesquisa.

¹² As entrevistadas se referem ao #INCLUDE como ferramenta, uma vez que nesta etapa da pesquisa ainda utilizávamos essa terminologia.

Consistência

Ao serem solicitadas a analisar o critério consistência, as entrevistadas também concordaram totalmente que o #INCLUDE está consistente com a proposta de construção de um aparato tecnológico com viés prático, que incentiva o desenvolvimento de atividades pedagógicas na matemática, aplicando os princípios do pensamento computacional. Ao justificarem tal percepção na escrita livre destacamos registros como: *“a ferramenta, da forma como está hoje, já possibilita o desenvolvimento de várias atividades na área de matemática”*; *“o #INCLUDE tem como ganho a simplicidade de aproximar o professor com formação em Pedagogia, onde os conceitos matemáticos não são ensinados porém são responsáveis pela alfabetização matemática dos alunos”*

Observa-se na análise das respostas a este critério o estímulo que o #INCLUDE promove nos professores pelas múltiplas possibilidades de desenvolvimento de atividades matemáticas em níveis distintos de formação e o incentivo à formação continuada dos professores pela especialização em matérias não aprofundadas na sua formação de graduação.

Praticidade

Novamente houve coerência na análise desse critério pelas entrevistadas, já que ambas concordaram totalmente que no #INCLUDE existe potencial à sua aplicabilidade pelo fato dele ser intuitivo, apresentar abstrações da realidade de atividades em sala de aula, o que facilita a adesão dos professores para a criação de intervenções pedagógicas no ensino da matemática. Nesse ponto de vista, registros como *“acredito que a escola não precise de tantos aparatos tecnológicos para o uso desta ferramenta”* e *“[#INCLUDE] possibilita a autonomia do professor em construir suas próprias atividades de acordo com a realidade dos seus alunos”*.

Ao analisar mais detalhadamente esse critério a por meio das percepções apresentadas observa-se outros potenciais não explicitamente almejados nos objetivos de concepção do #INCLUDE: a possibilidade de aplicação da modelagem matemática e da Etnomatemática.

Efetividade

Diferentemente dos critérios anteriores, houve concordância parcial das entrevistadas quando perguntadas se as funcionalidades disponíveis na #INCLUDE possibilitam que sejam

alcançados os resultados almejados ao produzir intervenções para o ensino de conceitos da matemática usando o pensamento computacional como estratégia pedagógica. Houve o registro da necessidade de ajustes técnicos relacionados a caixas de textos explicativos para a interface dos alunos visando fornecer auxílio à compreensão de funcionalidades #INCLUDE presentes na área de programação por blocos, a exemplo de “variável” e “texto”. Nesse critério, destacamos também o seguinte registro:

“Da maneira como a ferramenta está no momento, nem todos os conteúdos matemáticos dariam para ser trabalhados. Mas isto não vejo como falha. Vejo como um potencial a ser melhorado. Já me coloco à disposição para [...] fazermos uma equipe de alguns professores de matemática para junto pensarmos em outros comandos para a ferramenta que possibilitem uma maior abrangência dos conteúdos matemáticos”

Na análise da percepção nesse critério, ressaltamos o bom nível de aceitação de #INCLUDE por parte das entrevistadas ao sugerirem melhorias, tanto nos seus aspectos funcionais como na disponibilidade em contribuir na ampliação de sua abrangência. Entendemos que esse é um aspecto subjetivo muito positivo ao considerarmos que estamos tratando com a primeira versão do #INCLUDE.

Finalizamos esta seção de avaliação do #INCLUDE com um dos registros escritos da entrevista, que consideramos pertinente e que resume os ideais da concepção, planejamento e desenvolvimento alicerçado por uma metodologia de pesquisa deste projeto.

“A parabenizo por pensar no professor como uma peça fundamental para a entradas das tecnologias na Educação Básica. Ainda existe muita resistência de meus colegas, mas com uma ferramenta dessas, fácil de ser gerida, essa resistência pode ser quebrada. Me coloco a disposição se você quiser validar esta ferramenta em uma escola de Ensino Fundamental II.”

4.4 PRINCÍPIO DE DESIGN RESULTANTE

A partir da discussão apresentada em torno da avaliação do ambiente #INCLUDE, é evidenciado como princípio de design resultante deste ciclo da Pesquisa-aplicação, a necessidade de participação ativa do professor durante o processo de criação e ajuste de atividades que serão aplicadas utilizando ferramentas online. Esse princípio, é corroborado por seu destaque como uma das principais vantagens do #INCLUDE durante o seu último ciclo avaliativo, apontando a possibilidade da inclusão de questões, ou ajuste de entradas e saídas, de acordo com a realidade vivida em sala de aula por parte dos educadores.

Reforçamos que, conforme preconizado na metodologia de pesquisa adotada, o princípio de design resultante deste projeto de pesquisa promove o processo evolutivo dos trabalhos científicos na medida em que se torna um princípio de design inicial, a partir de sua divulgação, para propostas de intervenção de outros autores nessa temática.

5 CONSIDERAÇÕES FINAIS

O pensamento computacional evoluiu para uma temática de interesse nas questões associadas a diretrizes para a Educação Básica. A BNCC explicitamente o cita como habilidade a ser desenvolvida nos estudantes, em associação à educação matemática. Este trabalho de pesquisa foi desenvolvido com foco nesse tema, como forma de ampliar os estudos de questões e possibilidades de soluções nesse contexto, com especial direcionamento para a Rede Pública de Ensino.

Ao iniciarmos o trabalho, alguns objetivos foram definidos a partir de sua proposta, que buscou preencher a lacuna de pesquisa relacionada ao estudo de ferramentas de apoio com base tecnológica que promovessem o desenvolvimento do pensamento computacional, explorando às especificidades das matérias de formação no Ensino Básico e, nesse processo, buscasse também orientar os professores envolvidos para que eles próprios desenvolvessem essa competência a partir de uma atuação ativa.

Uma das primeiras questões nesse sentido, compreendeu a definição da disciplina de formação dos estudantes que se associaria com o desenvolvimento da habilidade do pensamento computacional, tendo como foco o projeto e a implementação de um ambiente tecnológico para este fim. Nesse quesito, a escolha da Matemática apresentou-se como uma decisão correta por suas características de intrinsecamente ligadas ao raciocínio lógico e estruturado na aplicação dos seus conceitos, fato que foi corroborado, no enfoque metodológico, a partir da avaliação de especialistas nessa área. O ambiente #INCLUDE surgiu com essas premissas e ao longo do seu desenvolvimento, associado à metodologia de pesquisa adotada, demonstrou que, além do atendimento à proposta, apresenta um potencial apontado pelas especialistas consultadas de expansão para conteúdos que abranjam desde o ensino fundamental até o nível médio, como previsto pela BNCC.

A programação visual por blocos do #INCLUDE, definida como um outro princípio de design para nortear a construção da ferramenta, assim como a matemática, também foi positiva. As especialistas consultadas acreditam que a principal qualidade desse tipo de programação, o enfoque na lógica, facilita a assimilação dos conceitos tanto do pensamento computacional quanto aqueles associados à Matemática. Além disso, foi denotado que esse modelo de

programação estimula a busca por diferentes soluções para um mesmo problema, uma vez que os estudantes podem montar os blocos que resolverão uma determinada questão utilizando os conceitos matemáticos aprendidos no lugar de aplicar fórmulas ou procedimentos prontos.

A avaliação das especialistas, nos seus dois ciclos, validou e enriqueceu o processo de criação e apresentação da proposta de solução em um nível multidisciplinar profissional. É importante ressaltar que a decisão de levar a proposta para uma avaliação qualitativa com profissionais especialistas, com graduações a nível de mestrado e doutorado e experiência em sala de aula na área, rendeu frutos que não se assemelhariam ao de uma análise apenas quantificável neste primeiro momento. Não obstante, o #INCLUDE é passível de validação em um meio escolar como trabalho futuro, essa afirmação só é possível graças ao parecer favorável nos dois ciclos avaliativos do enfoque metodológico aplicado. Ou seja, essa abordagem confirmou-se na prática como bem escolhida, uma vez que permitiu a utilização da metodologia da Pesquisa-aplicação tendo como ponto de corte os ciclos avaliativos com professores especialistas, atendendo, portanto, às restrições de tempo de fechamento de um projeto de Trabalho de Conclusão de Curso de graduação, ao passo que possibilita em um estágio de pesquisa futuro a aplicação com estudantes. Nesse sentido fica claro a ideia inicial de valorização do papel do professor na concepção de intervenções com vistas à melhoria dos processos de ensino.

A construção da solução guiada pela metodologia Pesquisa-aplicação possibilitaram o desenvolvimento de uma proposta de intervenção na Educação consistente e um processo de pesquisa fundamentado. Os questionamentos que naturalmente surgiram ao longo do caminho, diversas vezes foram sanados pela referencial teórico associado à metodologia de pesquisa que se mostrou elemento fundamental para guiar o trabalho científico.

Pessoalmente, trilhar esse caminho com a certeza de que havia um embase científico ao longo dele tornou essa experiência de formação produtiva, segura e ao mesmo tempo altamente gratificante, como expresso por Paulo Freire, nosso eterno Patrono da Educação, no seguinte trecho da epígrafe deste trabalho: “A alegria não chega apenas no encontro do achado, mas faz parte do processo da busca.”.

6 REFERÊNCIAS

- BNCC. Base Nacional Comum Curricular. **Base Nacional Comum Curricular**, 2017. Disponível em: <basenacionalcomum.mec.gov.br>. Acesso em: janeiro 2020.
- BOMBASAR, J. R. et al. Ferramentas para o Ensino-Aprendizagem do Pensamento Computacional: onde está Alan Turing? **Anais do XXVI Simpósio Brasileiro de Informática na Educação**, outubro 2015. 81-90.
- BUITRAGO FLÓREZ, F. et al. Changing a Generation's Way of Thinking: Teaching Computational Thinking Through Programming. **Review of Educational Research**, v. 87, n. 4, p. 834-860, 2017.
- EDUCATION CLOSET. What Is Steam. **Education Closet**, 2014. Disponível em: <<https://educationcloset.com/steam/what-is-steam/>>. Acesso em: 16 Julho 2019.
- EULER, D. Design principles as bridge between scientific knowledge production and practice design. **EDeR Educational Design Research**, 1, n. 1, 26 janeiro 2017.
- FACEBOOK INC. **React**, 2020. Disponível em: <reactjs.org>. Acesso em: 13 fevereiro 2020.
- GOOGLE DEVELOPERS. **Blockly**, 2020. Disponível em: <<https://developers.google.com/blockly/>>. Acesso em: 13 fevereiro 2020.
- HERRINGTON, J.; REEVES, T. Using design principles to improve pedagogical practice and promote student engagement. **Ascilite**, Hobart, 4-7 dezembro 2011. 600.
- HICKMOTT, D.; PRIETO-RODRIGUEZ, E.; HOLMES, K. A Scoping Review of Studies on Computational Thinking in K–12 Mathematics Classrooms. **Digital Experiences in Mathematics and Programming**, v. 4, p. 48-69, dezembro 2017.
- ISAACSON, W. **Os Inovadores: Uma biografia da revolução digital**. Tradução de Vários Tradutores. 1 Edição. ed. São Paulo: Companhia das Letras, 2014. 575 p. ISBN 9788535925029.
- ISRAEL, M. et al. Supporting all learners in school-wide computational thinking: A cross-case qualitative analysis. **Computers & Education**, v. 82, p. 263-279, 2015.
- MARTINELLI, S. R.; ZAINA, L. A. M.; SAKATA, T. **O Pensamento Computacional em Atividades de Ensino mediadas pelo Professor do Ensino Fundamental I: Um Estudo de Caso**. O Pensamento Computacional em Atividades de Ensino mediadas pelo PrUniversidade Federal de São Carlos. Sorocaba, p. 1-48. 2018. (O Pensamento Computacional em Atividades de Ensino mediadas pelo Professor do Ensino Fundamental I: Um Estudo de Caso. Relatório Técnico: 004/2018, Universidade Federal de São Carlos).
- MORAIS, A. D. D.; BASSO, M. V. D. A.; FAGUNDES, L. D. C. Morais, Anuar Daian de; Basso, Marcus Vinicius de Azevedo; Fagundes, Léa Da Cruz Educação Matemática & Ciência da Computação na escola: aprender a programar fomenta a aprendizagem de matemática? **Ciência & Educação**, Bauru, v. 23, n. 2, p. 455-473, junho 2017.

OPENJS FOUNDATION. **Node.js**, 2020. Disponível em: <<https://nodejs.org>>. Acesso em: 13 fevereiro 2020.

ORACLE CORPORATION. **MySQL**, 2020. Disponível em: <<https://www.mysql.com>>. Acesso em: 13 fevereiro 2020.

PESSOA, F. I. R. et al. T-mind: um Aplicativo Gamificado para Estímulo ao Desenvolvimento de Habilidades do Pensamento Computacional. **Anais do XXVIII Simpósio Brasileiro de Informática na Educação**, outubro 2017. 645-654.

PINHO, G. et al. Proposta de Jogo Digital para Dispositivos Móveis: Desenvolvendo Habilidades do Pensamento Computacional. **Anais do XXVII Simpósio Brasileiro de Informática na Educação**, novembro 2016. 100-109.

PLOMP, T. et al. **Pesquisa-Aplicação em Educação: uma introdução**. 1. ed. São Paulo: Artesanato Educacional, v. 1, 2018.

PRESSMAN, R.; MAXIM, B. **Engenharia de Software: Uma Abordagem Profissional**. 8a Edição. ed. Porto Alegre: AMGH, v. 1, 2015.

REEVES, T. Design Research from a technology perspective. **Educational design research**, London, 2006. 52-66.

SAEZ-LOPEZ, J.-M.; ROMAN-GONZALEZ, M.; VAZQUEZ-CANO, E. Visual programming languages integrated across the curriculum in elementary school: A two year case study using “Scratch” in five schools. **Computers & Education**, v. 97, p. 129141, 2016.

SENIN, S.; NASRI, N. M. Teachers’ Concern towards Applying Computational Thinking Skills in Teaching and Learning. **Teachers’ Concern towards Applying Computational Thinking International Journal of Academic Research in Business & Social Sciences**, v. 9, n. 1, p. 296–310, 2019.

VALENTE, J. A. Integração do pensamento computacional no currículo da educação básica: Diferentes estratégias usadas e questões de formação de professores e avaliação do aluno. **E-curriculum**, v. 14, n. 3, p. 864-897, 2016.

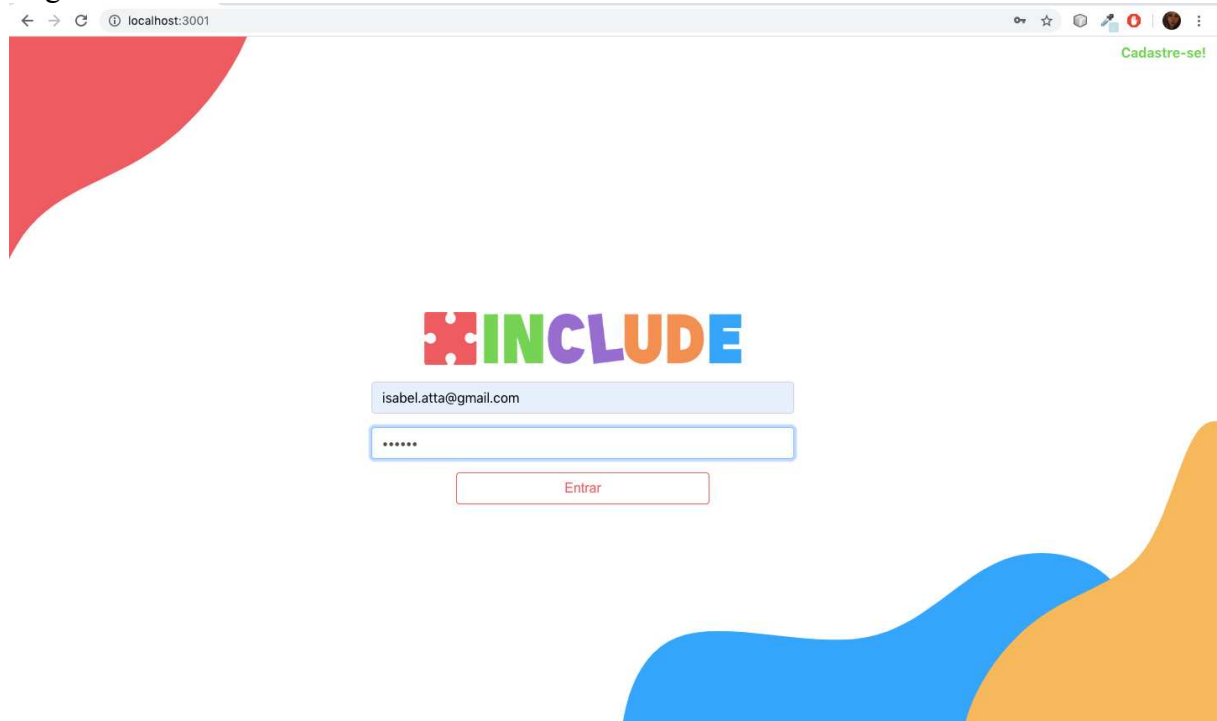
WING, J. M. Computational Thinking. **Communications of the ACM - Self managed systems**, v. 49, n. 3, p. 33-35, 2006.

WOHLGETHAN, E. Supporting Web Development Decisions by Comparing Three Major JavaScript Frameworks: Angular, React and Vue.js, 15 maio 2018.

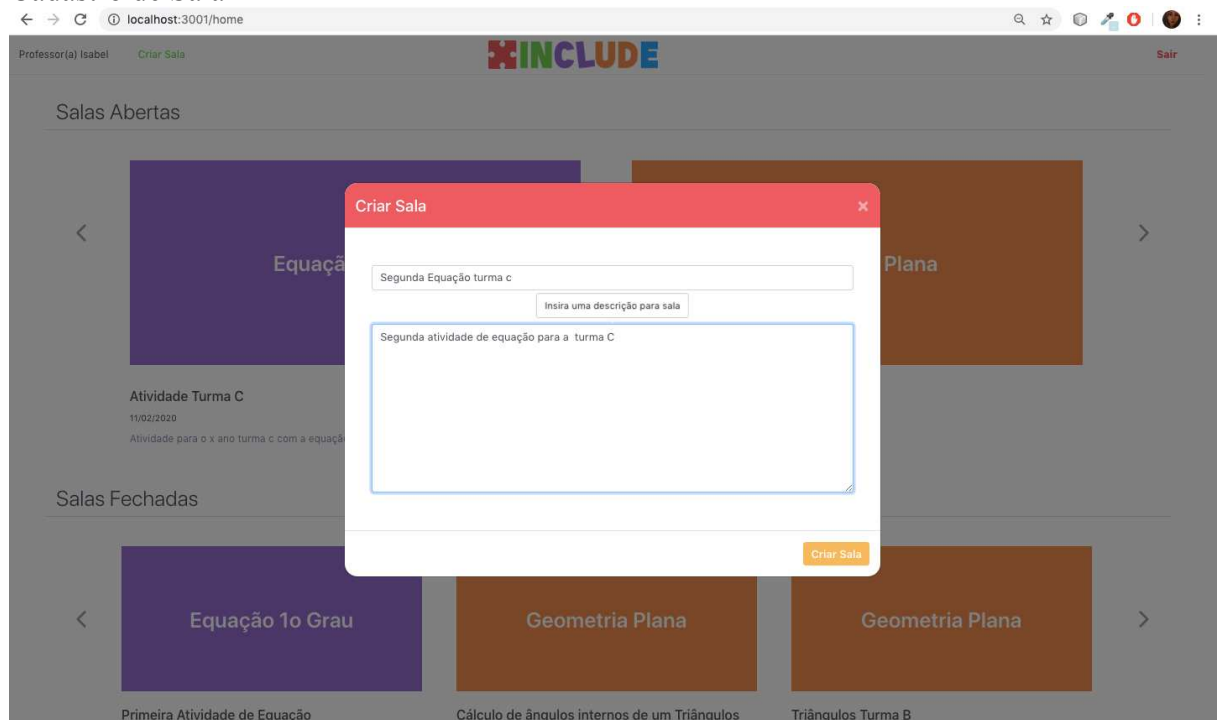
APÊNDICES

TELAS DO AMBIENTE #INCLUDE MENCIONADAS NO CAPITULO 4 SEÇÃO 3.3

Login de Usuário



Cadastro de Sala



Cadastro de Equipe ou Aluno

localhost:3001/sala

Segunda Equação turma c

Descrição da Atividade

Resolva a Equação $ax + b = 0$ para os determinados valores de a e b .

Exemplo de Saída:

Entradas

INCLUDE

Matemática
Variável
Texto
Lógica

Cadastro de Entradas e Saída

localhost:3001/criarAtividade

Criar Atividade

Escolha um assunto:

Equação 1o Grau

Geometria Plana

Grandezas Proporcionais

Escolha uma atividade:

Descrição da atividade:

Resolva a Equação $ax + b = 0$ para os determinados valores de a e b .

Entradas:

Informe as Entradas e Saídas esperadas

Modelo de Entrada: $a = W; b = y$

→

Modelo de Saída: $x = Z$

MODELO DE QUESTIONÁRIO APLICADO ÀS ESPECIALISTAS NO SEGUNDO CICLO AVALIATIVO



QUESTIONÁRIO DE AVALIAÇÃO FORMATIVA PEQUISA-APLICAÇÃO DE DESENVOLVIMENTO DA FERRAMENTA DE PENSAMENTO COMPUTACIONAL #INCLUDE

Pesquisadora: Isabel Valderrama Atta

Após a apresentação da ferramenta #INCLUDE como projeto de ambiente prático para intervenções que objetivem o desenvolvimento do pensamento computacional em estudantes a partir do ensino fundamental, analise essa proposta respondendo as seguintes perguntas.

1. A ferramenta apresenta uma abordagem diferenciada das propostas existentes para o ensino de matemática, a partir de intervenções com suporte tecnológico e incentivo ao desenvolvimento da habilidade do pensamento computacional na solução de problemas
 - () Discordo totalmente
 - () Discordo parcialmente
 - () Indiferente
 - () Concordo parcialmente
 - () Concordo totalmente

Por favor, forneça comentários que ofereçam uma compreensão qualitativa à sua resposta.

2. A partir do conhecimento sobre pensamento computacional apresentado, acredito que a ferramenta está consistente com a proposta de construção de um aparato tecnológico com viés prático, que incentive o desenvolvimento de atividades pedagógicas na matemática, a serem realizadas aplicando os princípios do pensamento computacional.

- Discordo totalmente
- Discordo parcialmente
- Indiferente
- Concordo parcialmente
- Concordo totalmente

Por favor, forneça comentários que ofereçam uma compreensão qualitativa à sua resposta.

3. Considerando as funcionalidades apresentadas e o exemplo de funcionamento da ferramenta é possível afirmar que existe potencial à aplicabilidade da mesma pelo fato dela ser intuitiva, apresentar abstrações da realidade de atividades em sala de aula, facilitado a adesão dos professores para criação de intervenções pedagógicas no ensino da matemática usando essa ferramenta (ao responder esta pergunta considere que a escola possui infraestrutura tecnológica adequada e que os professores têm formação e facilidade para o uso de tecnologia)

- Discordo totalmente
- Discordo parcialmente
- Indiferente
- Concordo parcialmente
- Concordo totalmente

Por favor, forneça comentários que ofereçam uma compreensão qualitativa à sua resposta.

4. As funcionalidades disponíveis na ferramenta possibilitam que sejam alcançados os resultados almejados ao produzir intervenções para o ensino de conceitos da matemática usando o pensamento computacional como estratégia pedagógica
- Discordo totalmente
 - Discordo parcialmente
 - Indiferente
 - Concordo parcialmente
 - Concordo totalmente

Por favor, forneça comentários que ofereçam uma compreensão qualitativa à sua resposta.

Use esta folha para qualquer outro registro que julgar pertinente.