

# Uma Ferramenta para Auxiliar o Desenvolvimento do Pensamento Computacional em Crianças

Arthur Santos, Felipe Oliveira, Glaucia Campos  
arthur\_ido@hotmail.com, {felipeoliveira, glauciamelissa}@uern.br  
Universidade do Estado do Rio Grande do Norte (UERN)  
Departamento de Ciência da Computação - Natal/RN

## RESUMO

O Pensamento Computacional (PC) tem sido aplicado como uma estratégia para a resolução de problemas por meio dos conceitos da Computação. Na educação, a aplicação do PC representa uma forma de disseminar informações e produzir conhecimento. É possível encontrar na literatura diferentes soluções que aplicam esse conceito. Porém, cada solução deve considerar fatores como o nível de escolaridade, o conteúdo abordado e o sistema de ensino (público/privado) devido aos custos gerados. Este trabalho apresenta o processo de modelagem/simulação de uma ferramenta (brinquedo de programar), baseada em Arduino, com a finalidade de auxiliar a aplicação das concepções do PC na Educação Infantil. Para comprovar o seu funcionamento correto, a arquitetura da ferramenta foi implementada nos simuladores Proteus e Tinkercad e foram realizadas várias simulações. Espera-se que a ferramenta possa ser construída e implementada, futuramente, em escolas carentes de recursos e soluções com esse propósito.

## CCS CONCEPTS

• **Social and professional topics** → Computing education.

## PALAVRAS-CHAVE

Pensamento Computacional, Arduino, Educação Infantil

## 1 INTRODUÇÃO

Segundo Vilaça [21], é possível afirmar que a tecnologia está presente na maioria dos aspectos da vida humana, mesmo que indiretamente. Como exemplo, pode-se observar que, por meio da tecnologia, simples atividades como escutar músicas, estudar ou até mesmo enviar documentos tornaram-se tarefas muito mais fáceis. Porém, o uso da tecnologia pode ir além, como servir de instrumento capaz de aumentar o poder cognitivo humano. Consequentemente, os conhecimentos básicos acerca da Computação devem ser considerados tão importantes quanto os conhecimentos básicos da Matemática, Física e Química para a sociedade contemporânea [20].

Neste contexto, tem sido empregado o termo Pensamento Computacional (PC) [22] que se refere a um conjunto de habilidades e competências comuns à área da Ciência da Computação, utilizados

para desenvolver a capacidade de resolver problemas em diferentes níveis de abstração, em qualquer área do conhecimento, e em qualquer fase da vida, explorando a criatividade e a construção do saber ao longo do tempo.

Em 2020, o Fórum Econômico Mundial publicou "*O relatório do futuro do trabalho*" [24] destacando as 15 habilidades essenciais para o profissional do futuro. Dentre elas, a resolução de problemas complexos, o pensamento crítico, a criatividade e a flexibilidade cognitiva são habilidades que podem ser desenvolvidas ao se trabalhar o Pensamento Computacional. Essas informações corroboram com as análises de Wing [23] que sinalizam que as competências do Pensamento Computacional devem ser desenvolvidas por todas as pessoas, e não apenas para quem escolhe carreiras ligadas à área da Ciência da Computação. Desde então, diferentes estratégias para aplicar esse conceito estão sendo estudadas e aplicadas em diferentes cenários [8].

De acordo com Valente [19], o Pensamento Computacional ainda é pouco explorado na Educação Básica, pois a maioria das aulas de Informática são voltadas para a utilização de *softwares* de escritório. Esse conteúdo é considerado defasado quando consideramos a necessidade de crescimento cognitivo da criança. Portanto, o ambiente escolar representa um espaço interessante para a inserção dos conceitos computacionais no processo de aprendizagem, uma vez que o ensino da Computação como ciência pode contribuir de forma positiva na formação do aluno, instigando o desenvolvimento do raciocínio lógico, pensamento crítico, tomada de decisão, capacidade de reconhecimento de padrões e resolução de problemas [3].

Como incentivo, em fevereiro de 2022, foi aprovado o parecer da Norma sobre a Computação na Educação Básica como complemento à Base Nacional Comum Curricular (BNCC) pelo Conselho Nacional de Educação (CNE) [7]. De acordo com essa norma, a Computação na Educação Básica pode ser visualizada em três eixos: o Pensamento Computacional, a Cultura Digital e o Mundo Digital. Também, compreendendo que é fundamental para o Brasil que conteúdos de Computação sejam ministrados na educação básica, a Sociedade Brasileira de Computação (SBC), contribuiu, por meio da Diretoria de Educação, com a elaboração do documento com as diretrizes para o ensino de Computação na Educação Básica [15] e organizou o primeiro evento sobre Educação em Computação (EduComp 2021). Os cursos de nível superior de Computação também têm contribuído com esse cenário inserindo disciplinas sobre o Pensamento Computacional em suas grades curriculares [18].

Embora muitas iniciativas em prol da temática estejam sendo realizadas, desenvolver soluções que estimulem a aplicação do Pensamento Computacional nas escolas deve considerar diferentes fatores como o conteúdo a ser abordado, o nível de escolaridade e o sistema de ensino (público ou privado, por questões financeiras).

Fica permitido ao(s) autor(es) ou a terceiros a reprodução ou distribuição, em parte ou no todo, do material extraído dessa obra, de forma verbatim, adaptada ou remixada, bem como a criação ou produção a partir do conteúdo dessa obra, para fins não comerciais, desde que sejam atribuídos os devidos créditos à criação original, sob os termos da licença CC BY-NC 4.0.

EduComp'23, Abril 24-29, 2023, Recife, Pernambuco, Brasil (On-line)

© 2023 Copyright mantido pelo(s) autor(es). Direitos de publicação licenciados à Sociedade Brasileira de Computação (SBC).

Também, é preciso identificar qual o melhor formato para essa ferramenta considerando as vertentes que indicam como trabalhar o Pensamento Computacional na Educação Infantil. Portanto, ainda é considerado um desafio a concepção de soluções que impulsionem a resolução de problemas por meio dos conceitos básicos da Computação.

Considerando esse contexto, o objetivo deste trabalho consiste em propor uma ferramenta, baseada em Arduino, e no formato de um brinquedo de programar, que possa auxiliar o professor em sala de aula na realização de atividades que instiguem o desenvolvimento do Pensamento Computacional nas crianças. A ferramenta está direcionada para a Educação Infantil. Além do funcionamento correto, este trabalho tem como requisitos identificar os componentes adequados para a construção da ferramenta, presumindo a idade do seu público-alvo e a escassez de recursos financeiros das escolas públicas. Convém ressaltar que o uso da ferramenta não se restringe às escolas públicas, podendo também ser aplicada em escolas privadas. Para comprovar a corretude no funcionamento, a arquitetura da ferramenta foi implementada nos simuladores Proteus e Tinkercad e foram realizadas simulações.

Este trabalho está dividido da seguinte maneira: a Seção 2 apresenta o *Background*; os Trabalhos Relacionados estão descritos na Seção 3; a Seção 4 descreve a Metodologia abordada. O Brinquedo de Programar (descrição da proposta) está detalhado na Seção 5; o processo de Simulação da Ferramenta é mostrado na Seção 6; na Seção 7 são apresentados os Resultados e, por fim, as Considerações Finais na Seção 8.

## 2 BACKGROUND

### 2.1 Pilares do Pensamento Computacional

Embora a literatura apresente diferentes conceitos para o termo Pensamento Computacional [14] [3], inicialmente, Wing [22] o definiu como um conjunto de habilidades e competências comuns à área da Computação que pode ser utilizado para estimular a capacidade de resolver problemas em diferentes níveis de abstração e em qualquer área do conhecimento. Associado a esse conceito, foram definidos os quatro pilares fundamentais à solução de problemas a partir do Pensamento Computacional:

- **Decomposição:** processo de divisão do problema em partes menores, possibilitando a resolução de problemas complexos de forma mais simples.
- **Reconhecimento de Padrões:** capacidade de identificar padrões que podem ser aplicados à resolução de diferentes problemas.
- **Abstração:** Filtragem da classificação dos dados, criando mecanismos que permitam separar apenas os elementos essenciais em determinado problema, ignorando detalhes irrelevantes
- **Algoritmos:** sequência de instruções concisa e sem ambiguidade para a resolução de problemas. A formulação de um algoritmo passa pelos processos de decomposição, reconhecimento de padrões e abstração, sendo o pilar que agrega todos os demais.

### 2.2 Ensino do Pensamento Computacional

De acordo com Raabe <sup>1</sup>, existem cinco vertentes que indicam como trabalhar o Pensamento Computacional na Educação Infantil e nos anos iniciais do Ensino Fundamental:

- Jogos de programar
- Scratch.jr
- Computação Desplugada
- Robótica Pedagógica
- Brinquedos de Programar

Os Jogos de Programar [12] são aqueles nos quais o jogador precisa solucionar problemas que necessitam de lógica algorítmica por meio de uma notação de programação. À medida que os problemas são solucionados, o jogo avança para novas fases permitindo a inserção de novos conceitos.

Scratch Jr [13] é uma ferramenta desenvolvida a partir do Scratch, porém, com uma série de recursos que permite a crianças ainda não alfabetizadas, e que estejam na educação infantil, desenvolverem suas primeiras histórias programáveis reforçando, assim, habilidades cognitivas e sociais.

A Computação Desplugada [3] tem como objetivo ensinar os fundamentos de computação de maneira lúdica, sem fazer uso de computadores ou equipamentos eletrônicos. As atividades são criadas utilizando materiais simples como lápis, papel e caneta.

A Robótica Pedagógica [19] utiliza-se de aspectos/abordagens da robótica industrial em um contexto a partir do qual atividades de construção, automação e controle dos dispositivos possibilitam a aplicação concreta de conceitos em um ambiente de ensino/aprendizagem.

O Brinquedo de Programar [4] permite que as crianças programem, por meio de botões, uma sequência de ações a ser reproduzida por um robô que pode assumir aparências diversas como: carros, animais e outras figuras representativas do imaginário infantil. As instruções estão normalmente relacionadas à movimentação e a repetição dos movimentos.

## 3 TRABALHOS RELACIONADOS

Diversas soluções foram desenvolvidas com o objetivo de ensinar programação para crianças. Por exemplo, o RoPE (Robô Programável Educacional) [11] é um brinquedo de programar que permite crianças, a partir dos 3 anos, a estabelecerem contato com o Pensamento Computacional. As crianças programam o robô por meio de botões direcionais que representam a trajetória que o brinquedo deve seguir. O RoPE caminha em cima de um tapete pedagógico proposto pelos professores até alcançar o destino programado pelas crianças. Dessa forma, auxilia no desenvolvimento de conceitos de lógica e matemática adequados à sua faixa etária. Porém, uma vez programado, o caminho não fica visível para as crianças até a sua execução completa.

Semelhante ao RoPE, a Primo Toys desenvolveu um robô codificador para crianças de 3 a 9 anos (Cubetto [9]). A diferença está na forma de interação entre as crianças e o robô que acontece por meio de uma mesa de programar, na qual são encaixadas peças de Lego para formar o caminho a ser seguido pelo robô em cima dos

<sup>1</sup> André Luís Alice Raabe, *Palestra: Pensamento Computacional na Educação e Metodologias Ativas*, Laboratório Digital Educacional, Youtube, outubro de 2020. Disponível em <<https://www.youtube.com/watch?v=cOJF7KaSFYs>> Acesso em Outubro de 2022.

tapetes pedagógicos. No entanto, além de não conseguir comprá-lo no Brasil, ainda tem um custo elevado (\$ 225,00).

O Bit-O corresponde a um outro brinquedo de programar desenvolvido pela Natal Makers [6]. Consiste em um *kit* para montar o seu próprio robô capaz de reconhecer cores, medir distâncias, além de ter conectividade com diversos aplicativos por meio de Bluetooth. Apresenta-se como uma solução para práticas educacionais, todavia, embora com custos mais baixos do que o Cubetto, ainda é uma alternativa muito difícil para as escolas adquirirem devido ao seu valor (R\$ 429,00).

O brinquedo de programar BeeBot [17], produzido pela empresa Terrapin, executa sequências de instruções criadas pelas crianças e programadas em um painel de botões que fica na sua parte superior. Os movimentos do brinquedo são mover para frente/trás e girar 90 graus para direita/esquerda e permitem que o mesmo movimente-se em cima de tapetes pedagógicos. Esse brinquedo foi utilizado em algumas pesquisas com crianças na Educação Infantil [1] [10] e os autores concluíram que o mesmo possui um problema bastante inconveniente, que é a necessidade de limpar a memória que contém o programa anterior pressionando sempre o botão “clear” antes de programar. Também, algumas crianças demonstraram dificuldade em compreender a ação da rotação do brinquedo, o que talvez indique que o mesmo não é adequado para esta faixa etária. Adicionado a esses problemas, temos o custo para sua aquisição (em torno de R\$ 800,00).

## 4 METODOLOGIA

Considerando a problemática definida como escopo para este trabalho, seguimos um conjunto de estratégias direcionadas à definição da proposta de ferramenta de apoio ao desenvolvimento do Pensamento Computacional em crianças.

Inicialmente, identificamos, a partir da leitura de diferentes trabalhos científicos, que atividades lúdicas contribuem significativamente com o processo de aprendizagem, principalmente em crianças [1]. Diante dessa definição inicial, ou seja, trabalhar com propostas lúdicas, conversamos com algumas pedagogas que nos sugeriram como público-alvo crianças em processo na Educação Infantil, considerando suas experiências de salas de aula.

A definição do público-alvo nos conduziu à reflexão sobre qual a melhor proposta considerando as cinco principais vertentes para trabalhar o Pensamento Computacional na Educação Infantil e nos anos iniciais do Ensino Fundamental. Novamente, com a contribuição das mesmas pedagogas, concluímos que o brinquedo de programar representa a melhor proposta, até por ser mais intuitivo. A literatura também salienta a importância de usar brinquedos de programar como recurso para trabalhar o pensamento crítico, e desenvolver o raciocínio lógico e a resolução de problemas [1]. Posteriormente, também foi preciso definir o conteúdo a ser abordado nesse brinquedo de programar. Então, concluímos que não seria interessante atrelarmos apenas um conteúdo específico, mas que qualquer temática poderia ser inserida ao brinquedo.

Uma prerrogativa do projeto é atender a um público carente de recursos. Então, decidimos propor uma solução de baixo custo que possa ser utilizada pelas escolas públicas. Por essa razão, escolhemos

como tecnologia o Arduino. Porém, antes mesmo de desenvolvermos a solução física, implementamos a modelagem conceitual e realizamos um conjunto de simulações para verificarmos a correteza do comportamento do brinquedo de programar. Para as simulações foram utilizados dois simuladores diferentes: Tinkercad e Proteus, sendo o primeiro utilizado para fins educacionais e o segundo pela academia e indústria. Baseado nessas estratégias, construímos a arquitetura do brinquedo de programar e realizamos diversas simulações a cerca da correteza de seu funcionamento. As próximas seções apresentam a arquitetura do brinquedo de programar, juntamente com a descrição dos seus componentes, funcionamento geral e a identificação dos componentes físicos a serem utilizados durante o seu desenvolvimento.

## 5 BRINQUEDO DE PROGRAMAR

A Figura 1 apresenta a arquitetura do brinquedo de programar disposta em um tabuleiro dividido em 3 componentes: o Painel de Controle, a Mesa de Programação e a Mesa de Visualização, além dos Baralhos de Cartas Pedagógicas.

O Painel de Controle é o componente responsável pela inicialização do brinquedo, para dar início ao caminho (sequência de botões) definido pela criança, a fim de alcançar o objetivo da atividade, além de mostrar visualmente, por meio de LEDs e música, se a resposta indicada pela criança está correta ou não. Na Mesa de Programação, a criança programa o caminho a ser seguido pressionando os botões de uma matriz para atingir o objetivo da atividade. A Mesa de Visualização é composta por LEDs que são ativados considerando o caminho definido pela criança na mesa de programação. Embora não faça parte do tabuleiro central, o Baralho de Cartas Pedagógicas contém um conjunto de imagens ou texto, em formato de cartas, com o conteúdo pedagógico.

### 5.1 Painel de Controle

Em sua composição, o Painel de Controle é formado pelos botões PLAY, RESET, ON/OFF e de RESPOSTA Certa/Errada. Neste componente, também é possível encontrar 3 LEDs, sendo amarelo para indicar se a ferramenta está ligada ou não, verde para resposta certa e vermelho para a resposta errada, assim como um *buzzer* para tocar músicas.

Apenas o professor interage com os botões ON/OFF e RESPOSTA, por ser ele quem verifica se a criança acertou ou não a resposta. Em caso de acerto, o professor pressiona o botão RESPOSTA Certa/Errada e um som com a música de parabéns é emitido, assim como o LED verde deve ficar ligado. Do contrário, o professor pressiona e segura o mesmo botão por 2 segundos e um som com uma música de erro deve ser emitido, assim como uma luz vermelha ligada indicando que a criança errou a resposta.

A criança pode interagir com esse componente apertando o botão PLAY e o RESET, pois é ela quem dá início ao caminho formado na Mesa de Programação (PLAY), podendo corrigir caso tenha pressionado algum botão errado (RESET). Este botão (RESET) também pode ser pressionado pelo professor.

### 5.2 Mesa de Programação

Na Mesa de Programação, é possível observar uma matriz de botões com tamanho 3x3. A escolha do tamanho da matriz está diretamente

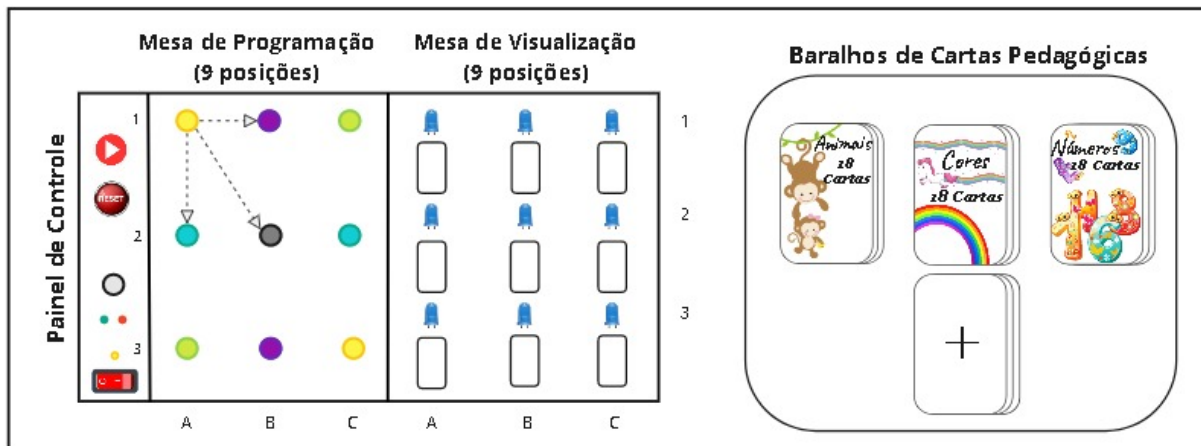


Figura 1: Arquitetura do Brinquedo de Programar

relacionada à escolaridade do público-alvo da solução (crianças no processo de alfabetização). A partir deste componente, a criança programa o caminho a ser seguido pressionando os botões em uma determinada sequência até chegar ao botão que representa a resposta final para ela (carta alvo na Mesa de Visualização). Obrigatoriamente, o ponto de partida é o primeiro botão do lado esquerdo. A partir desse botão, e pressionando os demais botões, a criança forma o caminho para chegar no objetivo, obedecendo as regras definidas para a ferramenta educacional (Tabela 1). Para uma melhor compreensão das regras, considere a matriz  $A(3 \times 3)$  composta pelos elementos  $a_{ij}$ , em que  $1 \leq i \leq 3$  representa a linha e  $A = \{j | 1 \leq j \leq 3$  a coluna.

Tabela 1: Regras para estabelecer caminhos na Mesa de Programação

REGRAS	EXEMPLOS
1. Após pressionar um botão, não é possível voltar ao botão anterior imediatamente pressionado	Considerando a matriz $M$ , se forem pressionados os botões $a^{1A}$ , $a^{1B}$ e, logo em seguida, o $a^{1C}$ , não é possível apertar o $a^{1B}$ novamente
2. Um novo botão somente pode ser pressionado caso esteja ligado diretamente com o botão atual, seja à sua frente, à sua diagonal ou abaixo	Considerando a matriz $M$ , se for pressionado o botão $a^{1A}$ , a criança somente pode pressionar os botões $a^{1B}$ , $a^{2A}$ e $a^{2B}$
3. Um caminho não pode conter o mesmo botão pressionado mais de uma vez (não forma ciclos)	Considerando a matriz $M$ , se forem pressionados os botões $a^{1A}$ , $a^{1B}$ , $a^{1C}$ , $a^{2C}$ , $a^{2B}$ , não é possível apertar o botão $a^{1B}$

### 5.3 Mesa de Visualização

A Mesa de Visualização também é composta por uma matriz de tamanho  $3 \times 3$  e um conjunto de LEDs (totalizando 9) para sinalizar o caminho especificado na Mesa de Programação. Somado a isso, na Mesa de Visualização também existem 9 espaços abaixo de cada LED, nos quais o professor encaixa as Cartas Pedagógicas com o

tema selecionado para a aula. Logo após o botão PLAY do Painel de Controle ser pressionado, os LEDs ligam e desligam na mesma ordem que os botões da Mesa de Programação foram pressionados. Assim, o último LED ligado indica a resposta da criança.

### 5.4 Baralhos de Cartas Pedagógicas

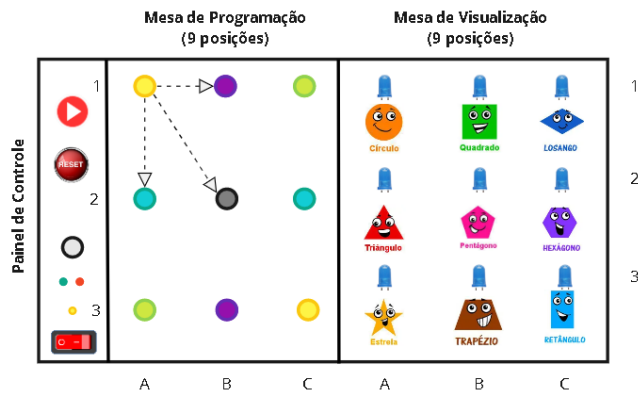
O conjunto de cartas com informações visuais sobre as temáticas foi chamado de Baralhos de Cartas Pedagógicas. O professor deve criar suas próprias cartas, mediante o assunto a ser abordado em cada aula (por exemplo, uma aula sobre as figuras geométricas). Essas cartas podem ser construídas com materiais simples e acessíveis às escolas como papel, lápis de cor, canetas, entre outros.

Para cada baralho (conteúdo abordado) são necessárias 18 cartas, sendo 2 de cada tipo (ou seja, existem 9 cartas diferentes, por exemplo), isso porque 9 cartas ficam distribuídas na Mesa de Visualização e as outras 9 ficam com os professores para o sorteio da carta alvo (objeto de estudo). Outro exemplo, é trabalhar com as 18 cartas diferentes, sendo 9 com perguntas (essas que ficam nas mãos dos professores) e as outras 9 com as respectivas respostas na Mesa de Visualização.

### 5.5 Funcionamento Geral

Inicialmente, a ferramenta deve ser ligada (LED fica amarelo) por meio do botão ON/OFF. O professor seleciona o assunto a ser abordado em sala de aula (ex. formas geométricas), distribui as cartas pedagógicas embaixo de cada LED da Mesa de Visualização (Figura 2) e mantém consigo as outras cartas (cópias das figuras).

A brincadeira pode ser realizada em pares ou mesmo o professor pode dividir a sala de aula em equipes para incentivar a participação das crianças, despertando o espírito competitivo e incentivar o trabalho em grupo. Nesse caso, uma criança de cada equipe precisa ser selecionada (por meio de um sorteio ou qualquer outra estratégia) para lançar um dado de 10 faces, que pode ser adquirido ou construído pelo professor a partir de uma planificação do sólido geométrico. O número que for obtido por meio do dado corresponde à posição da carta na mão do professor (o baralho de cartas deve ter sido embaralhado antes disso e não ficar visível inicialmente para



**Figura 2: Cartas de Figuras Geométricas distribuídas na Mesa de Visualização**

as crianças). Se o número sorteado for 7, a sétima carta que está na mão do professor deve ser a carta que a equipe precisa chegar na Mesa de Visualização, por exemplo, em um baralho de figuras geométricas, pode ter sido selecionada a carta do triângulo. Caso o número do dado seja 10, a equipe perde a oportunidade de descobrir onde está a carta alvo.

Cada equipe estabelece o caminho para chegar na carta alvo por meio da Mesa de Programação pressionando os botões conforme regras estabelecidas (Tabela 1). Depois do caminho ser construído, a criança sinaliza o final da sua programação por meio do Painel de Controle (botão PLAY) e, posteriormente, o caminho é mostrado na Mesa de Visualização mediante os LEDs vão sendo ligados, sendo que o último LED a ficar ligado corresponde a resposta da criança. Enquanto programa o seu caminho, a criança pode perceber que pressionou um botão errado e corrigir o caminho por meio do botão RESET que zera toda a programação. O professor deve, então, indicar se a criança acertou a resposta (pressiona o botão RESPOSTA e o LED verde liga) ou não (pressiona o botão RESPOSTA por 2 segundos e o LED vermelho liga), por intermédio da sinalização no Painel de Controle. Após a sinalização do professor, a ferramenta também emite sons diferentes para resposta correta ou errada.

A fim de tornar a brincadeira mais emocionante, o professor pode usar a ferramenta com regras de outros jogos, como exemplo, o jogo da forca no qual as crianças têm um número específico de tentativas para atingir um objetivo. Também pode fazer uso de um temporizador para identificar qual equipe conseguiu atingir o objetivo em menos tempo. Ou seja, o professor pode utilizar diferentes estratégias de Gamificação [16] a fim de tornar mais atrativo o aprendizado aos olhos das crianças. Para uma melhor compreensão do funcionamento do Brinquedo de Programar descrito nesta Seção, foi criado o fluxograma mostrado na Figura 3.

### 5.6 Identificação dos Componentes

Antes mesmo de desenvolver a ferramenta proposta, é preciso identificar os componentes necessários para a sua construção. Procurou-se indicar componentes financeiramente mais acessíveis à comunidade público-alvo, tendo em vista a precariedade de investimentos nas escolas públicas. Inicialmente, houve a preocupação com o tipo

de material a ser sugerido para evitar a ocorrência de acidentes. Também colocou-se empenho em identificar materiais que não se deterioram com facilidade. A Tabela 2 mostra quais foram os componentes e os seus respectivos dispositivos/materiais selecionados, assim como a justificativa para essas escolhas.

**Tabela 2: Identificação dos Componentes**

	COMPONENTES	DISPOSITIVOS / MATERIAIS	JUSTIFICATIVAS
Painel de Controle	Botões PLAY, RESET e de RESPOSTA Certa/Errada	Chave Tátil ou Push Button; Buzzer passivo	Preço mais acessível; possibilidade de ser customizado
	Botão LIGA/DESLIGA	Mini interruptor Chave Gangorra	Representa dois estados; controle do fluxo de corrente elétrica
Mesa de Programação	Matriz de Botões	Botões/ Chave Tátil	A criança consegue ter noção do caminho que ela está formando
Mesa de Visualização	Matriz de LEDs; Espaço para as Cartas Pedagógicas	LED difuso de alto brilho	É possível ver o caminho formado com as luzes coloridas até mesmo durante o dia
Cartas Pedagógicas	Baralho contendo o dobro de cartas em relação as posições da matrizes	Papel, Cartolina, Papelão, Fita, Coleção de Cor	Materiais encontrados em sala de aula, o educador pode customizar de acordo com o assunto
Tabuleiro Externo	Apoio e encaixe para os demais componentes	Medium Density Fiberboard (MDF)	Preço acessível; Resistente; Fácil de achar e customizar

A fim de limitar o fluxo de corrente e evitar a queima dos componentes utilizados, resistores devem ser adicionados à solução. Então, para os LEDs foram selecionados resistores de 470 ohms e para os botões os de 150 ohms. Para este trabalho, selecionamos o Arduino UNO sendo a justificativa pela escolha o seu baixo custo, facilidade de encontrar no mercado, assim como sua capacidade de receber diversos sensores diferentes. Também, por termos várias placas Arduino disponíveis no laboratório LUMEN, vinculado ao grupo de Pesquisa GSET/CNPQ da UERN - Campus de Natal.

Devido à quantidade limitada de portas do Arduino UNO, faz-se necessária a utilização de dois circuitos integrados 74HC595. Esses circuitos são do tipo registradores de deslocamento e funcionam como uma extensão das portas digitais do Arduino, sendo nesse projeto utilizados para o funcionamento dos LEDs. Para garantir que nenhum dado seja perdido no processo, deve ser utilizado um capacitor eletrolítico de 1uF por 50V. Ele age como um estabilizador, diminuindo a quantidade de variação da tensão enviada para os 74HC595.

Para auxiliar na montagem do circuito, deve ser utilizada uma matriz de contato chamada de protoboard. Ela permite que os circuitos eletrônicos sejam montados sem a necessidade de soldagem dos componentes. A partir dessa placa, é possível aumentar o número de portas de alimentação do Arduino como por exemplo o +5V e o Ground (GND), permitindo que mais componentes, como sensores, circuitos integrados e outros sejam alimentados. Para este projeto, pode ser usada uma protoboard de 830 pontos.

A Tabela 3 apresenta os preços dos componentes utilizados para construir o brinquedo de programar. Estes preços foram coletados em Setembro/2022, no aplicativo da Shopee. Todavia, esses valores não consideram o valor cobrado pelo frete dessas mercadorias. Também não foram considerados valores referentes à aquisição de outros materiais como chaves de fenda, estilete, cola, ferro de solda, fios, entre outros, uma vez que são materiais que comumente as pessoas dispõem em casa.

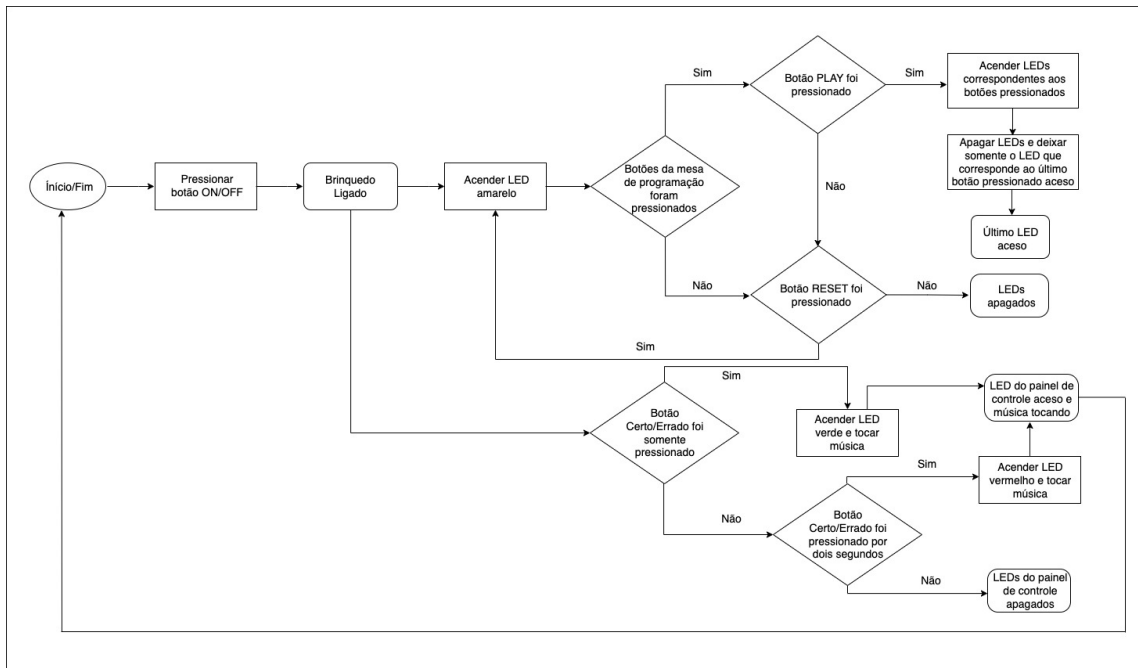


Figura 3: Fluxograma com o funcionamento geral do Brinquedo de Programar

Tabela 3: Preços dos Componentes

COMPONENTE	QUANTIDADE	VALOR (Shopee)
Arduino UNO	1	R\$ 59,90
Protoboard 830 pontos	1	R\$ 9,90
LEDs de alto brilho	12	R\$ 1,74
Resistores	22	R\$ 1,54
Capacitor Eletrolítico	1	R\$ 1,00
Circuito integrado (74HC595)	2	R\$ 2,00
Buzzer passivo	1	R\$ 2,50
Botões	12	R\$ 12,00
Chave gangorra	1	R\$ 1,50
MDF	2	R\$ 16,00

para o modelo esquemático. Adicionalmente, foram gerados arquivos de texto com o histórico dos eventos da simulação como forma de comprovar o seu funcionamento.

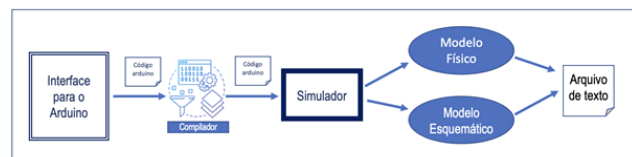


Figura 4: Processo de simulação

## 6 SIMULAÇÃO

Para garantir a corretude do comportamento do brinquedo de programar, foram utilizados dois simuladores (Tinkercad [2] e o Proteus [5]), sendo o primeiro utilizado para fins educacionais e o segundo pela academia e indústria. O propósito deste trabalho não consiste em fazer um comparativo entre essas ferramentas de simulação. A escolha por dois simuladores está associada ao produto que pode ser gerado por cada uma dessas ferramentas, o que torna a verificação mais completa.

A ideia geral do processo de simulação realizado pelas duas ferramentas é apresentada na Figura 4. Todas essas ferramentas precisam de um código Arduino a ser criado em uma interface de usuário, depois compilado e, posteriormente, simulado. O resultado da simulação corresponde a um modelo físico ou esquemático dependendo da ferramenta, sendo o *Tinkercad* para o modelo físico e o *Proteus*

A Figura 5 mostra o diagrama esquemático completo gerado a partir da ferramenta proposta pelo Proteus. Na parte superior direita, estão os LEDs da Mesa de Visualização representados pela letra D de diodo e o seu número (ex. D1). Também existe um resistor associado a cada um deles para garantir que a corrente elétrica passada para o LED não seja maior do que ele suporta e venha a queimar. Logo abaixo, estão os 9 botões da Mesa de Programação demonstrados por botões/chave táctil. Para cada botão, tem-se um resistor também para fazer o controle da corrente elétrica, assim como nos LEDs.

Do lado esquerdo, é possível verificar os componentes do Painel de Controle. Os LEDs também representados pela letra D (diodo) com os seus respectivos resistores (R), além dos botões PLAY, RESET e RESPOSTA demonstrados por botões/chave táctil e seus resistores. O botão ON/OFF que permite a passagem de corrente para o funcionamento do Arduino e, consequentemente, de todo o circuito, descrito em 5.5, não é necessário existir na simulação do Proteus, pois o circuito a ser simulado é "ligado" através de um comando



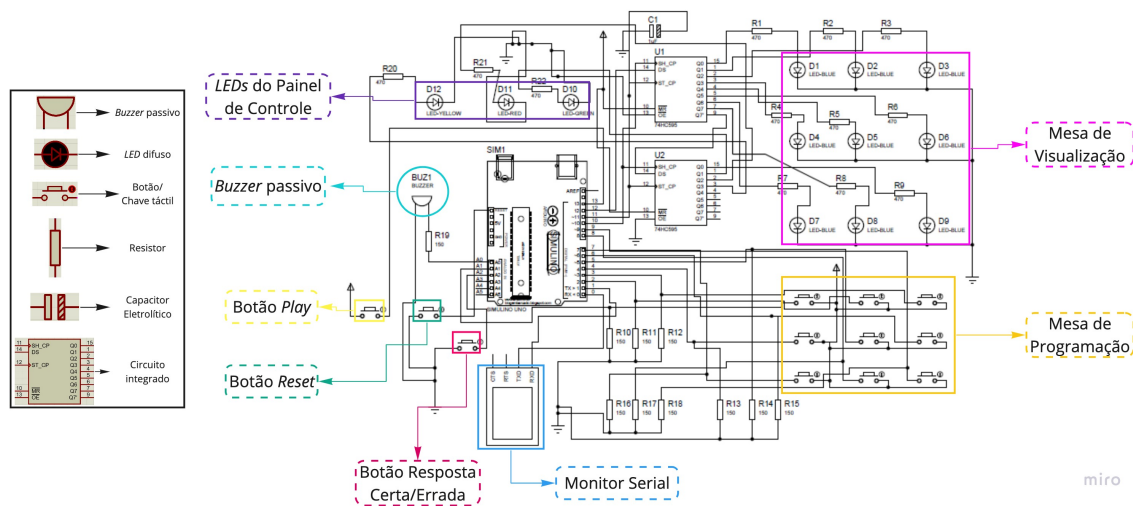


Figura 5: Simulação com o Proteus

próprio de execução pertencente ao Proteus. O buzzer passivo está representado pela palavra BUZ, sendo necessário para emitir o som associado a cada resposta. Para tornar possível a execução com 9 LEDs, 9 botões e a expansão para matrizes maiores, foram utilizados dois circuitos integrados 74HC595 que servem para expandir a quantidade de portas do Arduino, sendo ambos representados pela letra U seguida pelo seu número. O capacitor (C1) funciona como um estabilizador de tensão, evitando que aconteça a perda dos dados quando passados entre os circuitos integrados e o Arduino.

No código Arduino, inicialmente é feita a declaração das variáveis utilizadas para o funcionamento da ferramenta, como o estado do botão e o estado dos LEDs, por exemplo, deixando todos em 0, assim como a declaração da função que toca a música de resposta certa/errada. Logo em seguida, são declarados os botões como sendo *input* e os LEDs como *output*, deixando todos eles desligados, com exceção do LED amarelo que fica ligado para indicar o funcionamento da ferramenta. Então, é feita a verificação de quais os botões da Mesa de Programação foram pressionados, mudando o valor do seu estado para 1. Por exemplo, se o botão 2 for pressionado, o estado do botão muda de 0 para 1, deixando salva essa informação para ser utilizada em um passo mais adiante do programa. Logo depois, verifica-se o botão PLAY mudando o seu estado para 1, caso tenha sido pressionado.

Posteriormente, entra-se em um laço para a ativação dos LEDs da Mesa de Visualização. Esse laço é constituído por várias condições que verificam se o botão PLAY foi pressionado, se o botão correspondente aquele LED tem o seu estado em 1 e se o LED que está ligado a ele foi também pressionado. Caso satisfaça todas essas condições, o LED em questão é ligado. Por exemplo, para ativar o LED 3, é feita a verificação se o botão PLAY foi pressionado, logo depois, se o estado do botão 3 está em 1 e por fim, se o LED 2 tem seu estado ativado. Por último, caso o botão de RESPOSTA Certa/Errada tenha sido pressionado, é invocada a função que toca a música de parabéns e o LED verde muda o seu estado de desligado para ligado. Se for pressionado por 2 segundos, a função toca uma sirene e o LED vermelho é ligado junto à música. A Figura 6 apresenta um

trecho do código Arduino que indica o funcionamento da solução quando a criança pressiona o botão PLAY. Convém ressaltar que esse código Arduino é utilizado nas duas ferramentas de simulação.

```
//Liga os Leds após o botão play ser pressionado
if(digitalRead(botooplay) == HIGH){//Verifica se o botão play foi pressionado
if(estadole1 == 1 && anterior == 1){ //Verifica se o estado do LED 1 é para ser alto
Serial.println("Botao Play foi pressionado");
ci74HC595write(0, HIGH);//chama a função para acender o primeiro LED
Serial.println("LED 1 Acendeu");
anterior = 2;//Atualiza o valor para saber se o LED foi acesso
delay(300);
}
if(estadole2 == 1 && anterior == 2){//LED 2 só acende se o primeiro acender
ci74HC595write(1, HIGH);//Acende o LED atual
Serial.println("LED 2 Acendeu");
anterior = 3;// Atualiza o valor para saber se o LED foi acesso
delay(300);
ci74HC595write(0, LOW);//Apaga o LED anterior
Serial.println("LED Anterior Apagou");
}
if(estadole3 == 1 && anterior == 3){//LED 3 só acende se o LED 2 acender
ci74HC595write(2, HIGH);
Serial.println("LED 3 Acendeu");
anterior = 4;
delay(300);
ci74HC595write(1, LOW);
Serial.println("LED Anterior Apagou");
}
}
```

Figura 6: Trecho do código Arduino

A Figura 7 apresenta o modelo físico da solução gerado pelo Tinkercad. Esse modelo é formado por componentes mais próximo àqueles reais (ex, um LED no Tinkercad é igual ao LED físico). Porém, o Tinkercad corresponde a uma aplicação que funciona em um navegador da Internet, o que significa que o tempo de execução depende da qualidade da Internet utilizada. Por outro lado, o Proteus funciona no próprio sistema operacional, podendo-se obter um tempo de execução muito próximo ao real. Assim como descrito na simulação com o Proteus, a chave ON/OFF não é necessária no Tinkercad, pois ela é substituída pelo comando que inicia a simulação no Tinkercad, correspondendo a passagem de tensão para o Arduino.

## 7 RESULTADOS

Os resultados deste trabalho são apresentados seguindo duas diferentes *threads*. A Seção 7.1 mostra como foram colhidos os dados

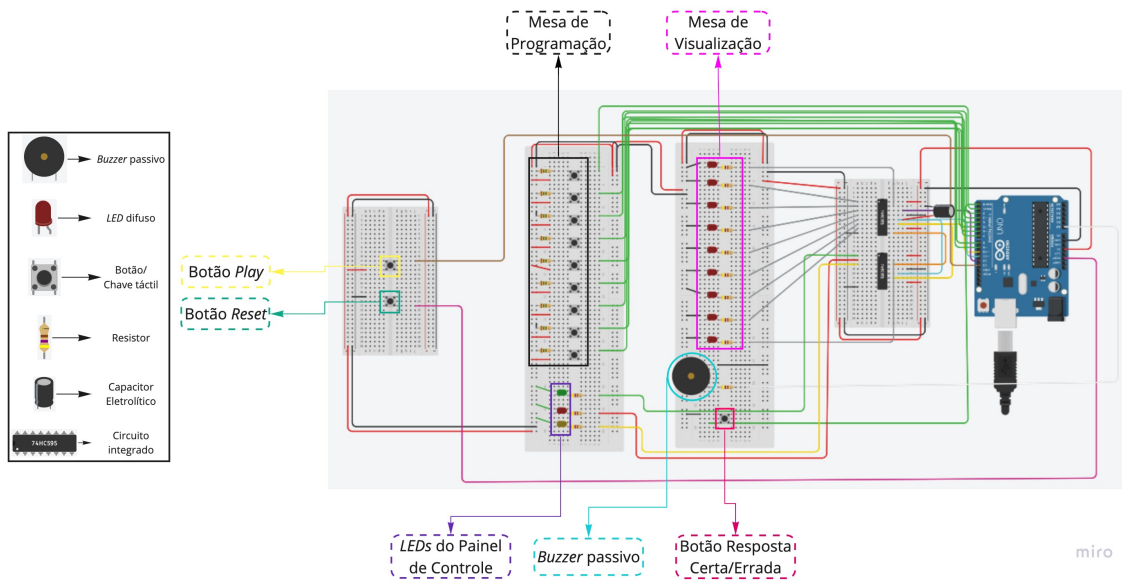


Figura 7: Simulação com o Tinkercad

para averiguarmos a corretude do comportamento do brinquedo de programar a partir das duas ferramentas de simulação. Na Seção 7.2, temos a relação entre o brinquedo de programar proposto e os pilares do Pensamento Computacional.

### 7.1 Simulação

Após criar os modelos físico e esquemático da solução proposta (brinquedo de programar), foram realizadas simulações a fim de comprovar a corretude no comportamento de cada modelo. Os resultados mostram que os modelos correspondem à proposta da ferramenta. Considerando que seria dispendioso mostrar o processo de simulação da ferramenta por meio de figuras sequenciais, os resultados estão disponíveis no formato de vídeos <sup>1 2</sup>.

No entanto, com o propósito de acompanhar o comportamento dos modelos em cada ferramenta, foram gerados arquivos de log (Padrão ASCII). Para isso, primeiro foi ativado o recurso de Monitor Serial do Arduino nas duas ferramentas (Tinkercad e Proteus) (Código 1). Com este recurso, é possível visualizar informações sobre os eventos realizados durante a simulação (ex. LED amarelo ligado).

#### Código-fonte 1: Monitor Serial em Arduino

```
//habilita o monitor serial
Serial.begin(9600);
//imprime no monitor
Serial.println("LED Amarelo Ligado");
//imprime no monitor
Serial.println("Brinquedo pronto para novo caminho");
```

Como exemplo, a Figura 8 representa o log gerado quando a criança estabelece o caminho correto para alcançar o objetivo da atividade. É possível observar o conjunto de eventos gerados, inclusive o momento de início e término da atividade.

```
LED Amarelo Aceso
Brinquedo pronto para novo caminho:
Botao 1 foi pressionado
Botao 2 foi pressionado
Botao 3 foi pressionado
Botao Play foi pressionado
LED 1 Acendeu
LED 2 Acendeu
LED Anterior Apagou
LED 3 Acendeu
LED Anterior Apagou
Resposta Certa :D
LED Verde Acendeu
Musica tocando
Musica parou de tocar
LED Verde Apagou
```

Figura 8: Arquivo de Log

### 7.2 Relação entre a Ferramenta e os Pilares do Pensamento Computacional

É possível ressaltar que houve uma preocupação em contemplar os pilares que compõem o Pensamento Computacional na ferramenta proposta como um dos resultados deste trabalho.

**7.2.1 Decomposição.** Processo que divide os problemas em partes menores para facilitar a resolução, desenvolvimento e gerenciamento. Pode ser visualizado em diferentes cenários, como:

- Considere a criança formando palavras a partir de Cartas Pedagógicas que contém as letras do alfabeto ou sílabas. Dessa forma, o professor consegue atingir um objetivo maior ao quebrar as palavras em sílabas e essas em letras fazendo com que a criança resolva o problema por partes.
- A criança precisa reconhecer na Mesa de Visualização onde está a carta com o objeto (ex. figuras, cores) selecionado. Depois, necessita identificar quais são os possíveis caminhos

<sup>1</sup><https://bit.ly/3p95n12> (Tinkercad)

<sup>2</sup><https://bit.ly/3wLvRs2> (Proteus)



para se alcançar a carta objetivo. Por último, precisa escolher qual o melhor caminho para ela. Ou seja, o problema foi decomposto em três problemas menores.

**7.2.2 Abstração.** Processo que concebe mecanismos que permitam separar os elementos essenciais em um problema, ignorando detalhes irrelevantes. Pode ser reconhecido em diferentes situações:

- O professor consegue trabalhar com conteúdos que demandam detalhes abstraindo algumas informações. Por exemplo, para ensinar as letras do alfabeto, não precisa identificar diferentes tipos de fontes nas cartas pedagógicas.
- Para a criança alcançar a carta almejada, não precisa se preocupar com o conteúdo específico das cartas que formam o caminho até o destino.

**7.2.3 Reconhecimento de Padrões.** Processo de identificar similaridades ou características que alguns problemas compartilham. Pode ser observado em algumas situações:

- Quando a criança tem a capacidade de aproveitar caminhos conhecidos para chegar em uma nova resposta. Por exemplo, a posição 5 foi identificada pela criança anteriormente e na próxima rodada ela precisa chegar na posição 7. É possível utilizar o caminho que foi aprendido para alcançar a posição 5 e somente completar até a nova posição.
- A criança pode observar que sempre o professor coloca na Mesa de Visualização uma figura geométrica após um animal e fazer uso dessa informação para novas buscas.

**7.2.4 Algoritmos.** Representa uma sequência finita de etapas (passos) para resolver um problema específico. É possível identificar esse conceito durante toda a dinâmica, desde seguir os passos para a brincadeira, lançar os dados, pegar a carta sorteada, identificar a carta objetivo na Mesa de Visualização, entre outros, principalmente fazendo a construção do caminho para a solução na Mesa de Programação.

## 8 CONSIDERAÇÕES FINAIS

Este trabalho teve como resultado a proposta de uma ferramenta no formato de um brinquedo de programar, baseada em Arduino, para auxiliar no desenvolvimento do Pensamento Computacional de crianças que estão no processo de alfabetização (Educação Infantil). Inicialmente, foi apresentada a arquitetura da ferramenta e descritos individualmente os seus componentes, assim como o seu funcionamento. Posteriormente, foram identificados os componentes físicos que podem ser utilizados para uma implementação considerando as restrições do público-alvo.

Para avaliar a solução proposta, foram utilizados dois simuladores (*Tinkercad* e *Proteus*), sendo o primeiro comumente aplicado para fins educacionais e o segundo pela academia/indústria. Considerando as simulações realizadas, a ferramenta comportou-se corretamente de acordo com a arquitetura proposta, o que demonstra indícios de que uma vez implementada, a ferramenta pode atingir o seu propósito de contribuir com o desenvolvimento do Pensamento Computacional nas crianças ao ser utilizada nas escolas.

Como trabalhos futuros, identificou-se a necessidade da construção física da ferramenta e a sua validação junto às escolas. Também é possível aumentar o tamanho das matrizes de botões e LEDs para abordar conteúdos mais complexos, bem como implementar novos

caminhos com o objetivo de tornar possível a realização de voltas/ciclos nos caminhos para séries mais elevadas, aumentando a dificuldade da brincadeira.

## REFERÊNCIAS

- [1] Paula Angerami, Andre Raabe, and Tatiane Rosário. 2022. As aprendizagens desenvolvidas por crianças com o uso de brinquedos de programar. *Dialogia*.
- [2] Autodesk. 2021. Tinkercad. <http://www.tinkercad.com>. An optional note.
- [3] Christian Brackmann. 2017. *Desenvolvimento do Pensamento Computacional através de Atividades Desplugadas Básica*. Ph.D. Dissertation. Universidade Federal do Rio Grande do Sul (UFRGS).
- [4] Tatiane Aparecida Martins do Rosário. 2017. *As Aprendizagens Com O Uso Do Brinquedo De Programar*. Ph.D. Dissertation. Universidade do Vale do Itajaí, Itajaí, SC, Brasil. Advisor(s) André Luís Alice Raabe.
- [5] Labcenter Electronics. 2021. Proteus. <https://www.labcenter.com/simulation/>.
- [6] Natal Makers. 2022. Bit-O: Kit Educacional de Robótica e IoT. <https://www.natalmakers.com/loja/en/home/460-.html>.
- [7] Ministério da Educação. 2022. Normas sobre Computação na Educação Básica – Complemento à Base Nacional Comum Curricular (BNCC). <http://portal.mec.gov.br/>.
- [8] V. F. Moretti. 2019. O pensamento computacional no ensino básico : potencialidades de desenvolvimento com o uso do scratch. Monografia (Licenciado em Matemática), UFRGS (Universidade Federal do Rio Grande do Sul), Porto Alegre, Brasil.
- [9] Primo Toys. 2022. Cubetto Classic. <https://www.primotoys.com/>.
- [10] André Raabe, Alzira Rodrigues, André Santana, Marli Vieira, Tatiane do Rosário, and Ana Carneiro. 2015. Brinquedos de Programar na Educação Infantil: Um estudo de Caso. In *Anais do XXI Workshop de Informática na Escola*. SBC, Porto Alegre, RS, Brasil, 42–51.
- [11] Andre Raabe, Tatiane Rosário, Rodrigo Martins, Andre Santana, Felipe Souza, and Raphael Silva. 2017. RoPE - Brinquedo de Programar e Plataforma de Aprender. 1119.
- [12] Andre Raabe, Guilherme Zanchett, and Adilson Vahldick. 2015. Jogos de Programar como uma Abordagem para os Primeiros Contatos dos Estudantes com a Programação. 1485.
- [13] Andre Santana, Julia Metzger, Andre Raabe, Luis Santana, Alisson Henrique, Clara Maciel, and Annelize Maciel. 2017. Scratch Jr - Brincando em Família: um relato de experiências sobre introdução a programação no ambiente escolar. 432.
- [14] Bianca Santana, Christina Chavez, and Roberto Bittencourt. 2021. Uma Definição Operacional para Pensamento Computacional. In *Anais do Simpósio Brasileiro de Educação em Computação (On-line)*. SBC, Porto Alegre, RS, Brasil, 93–103.
- [15] Sociedade Brasileira de Computação SBC. 2021. Diretrizes para ensino de Computação na Educação Básica. <https://www.sbc.org.br/educacao/diretrizes-para-ensino-de-computacao-na-educacao-basica/>. An optional note.
- [16] Rodrigo Smiderle, Sandro José Rigo, Leonardo B. Marques, Jorge Arthur Peçanha de Miranda Coelho, and Patricia A. Jaques. 2020. The impact of gamification on students' learning, engagement and behavior based on their personality traits. *Smart Learning Environments* 7, 1.
- [17] Terrapin. 2022. Tools for thinking: Bee-Bot. <https://www.terrapinlogo.com/bee-bot-ss.html>.
- [18] Universidade Federal Rural de Pernambuco UFRPE. 2021. Licenciatura Plena em Computação - Matriz Curricular. <http://www.lc.ufirpe.br/br/matriz-curricular>. An optional note.
- [19] José Armando Valente. 2016. 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* 14, 864 – 897.
- [20] P. C. Venturini. 2020. Desenvolvimento do pensamento computacional por meio da Ciência da Computação Desplugada e do Scratch. *Revista Em Extensão* 18, 2, 200–208.
- [21] M. L. C. Vilaça and E. V. F. de Araújo. 2016. *Tecnologia, Sociedade e Educação na Era Digital* (1st ed.). Duque de Caxias, RJ: Unigranrio.
- [22] Jeannette M. Wing. 2006. Computational Thinking. *Commun. ACM* 49, 3, 33–35.
- [23] Jeannette M. Wing. 2010. Computational Thinking: What and Why? *The link Magazine* 6, 20–23.
- [24] World Economic Forum. 2020. The Future of Jobs Report. <https://www3.weforum.org/>. An optional note.