

CESAR PEREIRA VIANA

ROPE AR

UMA INTERFACE DE REALIDADE AUMENTADA PARA DEPURAÇÃO DE
ALGORITMOS POR CRIANÇAS DE 4 A 6 ANOS

Itajaí (SC), Setembro de 2021



UNIVALI

UNIVERSIDADE DO VALE DO ITAJAÍ

**PRÓ-REITORIA DE PÓS-GRADUAÇÃO,
PESQUISA, EXTENSÃO E CULTURA**

**PROGRAMA DE MESTRADO ACADÊMICO EM
COMPUTAÇÃO APLICADA**

ROPE AR

**UMA INTERFACE DE REALIDADE AUMENTADA PARA DEPURAÇÃO DE
ALGORITMOS POR CRIANÇAS DE 4 A 6 ANOS**

por

Cesar Pereira Viana

Dissertação apresentada como requisito parcial à
obtenção do grau de Mestre em Computação Apli-
cada.

Orientador: André Luis Alice Raabe, Dr.

Itajaí (SC), Setembro de 2021

“INTERFACE DE REALIDADE AUMENTADA PARA O BRINQUEDO PROGRAMÁVEL ROPE”

Cesar Pereira Viana

Esta Dissertação foi julgada adequada para obtenção do Título de Mestre em Computação Aplicada, Área de Concentração Computação Aplicada e aprovada em sua forma final pelo Programa de Mestrado Acadêmico em Computação Aplicada da Universidade do Vale do Itajaí.



Prof. André Luís Alice Raabe, Dr.
Coordenador do Curso de Mestrado em Computação Aplicada

Apresentado perante a Banca Examinadora composta pelos Professores:



Prof. André Luís Alice Raabe, Dr.
Orientador

Rudimar Luís Scaranto Dazzi
Prof. Rudimar Luís Scaranto Dazzi, Dr. (UNIVALI)
Membro

Prof.ª Lucia Maria Martins Giraffa Dra. (PUCRS)
Membro

Vânia Paula de Almeida Neris
Prof.ª, Vânia Paula de Almeida Neris Dr. (UFSCAR)
Membro

Pai e mãe. Meus professores para a vida.

“I believe with Dewey, Montessori and Piaget that children learn by doing and by thinking about what they do. And so the fundamental ingredients of educational innovation must be better things to do and better ways to think about oneself doing these things.”

Seymour Papert, 1972.

AGRADECIMENTOS

Ao meus pais. Sem vocês eu nada seria.

Ao meu irmão, companheiro e exemplo. Admiro muito teu trabalho e humildade. Você também foi fundamental neste trabalho.

Ao meu sobrinho Pedro, pelas opiniões sinceras, e também à minha irmã.

Ao professor André Raabe, pelas portas abertas, e por plantar muitas sementes de Educação. Que elas se expandam e transformem as salas de aula em locais de construção de conhecimento.

À Renate Raabe, pela oportunidade de participar no projeto Lite Is Cool. Este foi meu verdadeiro estágio de docência, experiência que vou levar pra vida. Aproveito e agradeço as estudantes, e ao meu colega Ivan. Creio que formamos uma bela equipe.

Ao meu amigo Noschang, que todos chamam de mago, pelos seus poderes computacionais. Obrigado pelos ensinamentos.

Ao Paulo, por compartilhar conhecimento e apoiar nas questões de hardware do brinquedo RoPE.

À Maraysa, secretária do MCA, por enviar o meu resumo para tradução quando eu esqueci, e por sempre incentivar e apoiar os estudantes.

À Sylvana, diretora do CDI, por todo apoio. Você foi fundamental para esse trabalho. Agradeço também à Secretaria Municipal de Educação de Gaspar.

Às professoras do CDI, pelas contribuições à pesquisa. Só posso dizer que gostaria de ser uma de suas crianças, brincando e aprendendo nos espaços que vocês constroem.

Às crianças participantes. Espero que tenham se divertido. Vocês foram a parte mais importante desse trabalho.

À Aline, minha companheira. Obrigado pelo apoio, pela paciência, e por entender os momentos de ausência. Com metade da sua agilidade essa dissertação estaria pronta em 1 ano.

Também ao Flick, meu companheiro, muitas vezes dormindo no meu colo enquanto eu escrevia esse texto.

Àos muitos professores que tive.

Àos integrantes do Lite e do GIE.

E, por fim, à CAPES e à Univali, pela bolsa de pesquisa.

ROPE AR

UMA INTERFACE DE REALIDADE AUMENTADA PARA DEPURAÇÃO DE ALGORITMOS POR CRIANÇAS DE 4 A 6 ANOS

Cesar Pereira Viana

Setembro / 2021

Orientador: André Luis Alice Raabe, Dr.

Área de Concentração: Computação Aplicada

Linha de Pesquisa: Informática Aplicada ao Ensino de Computação

Palavras-chave: Depuração, realidade aumentada, brinquedos programáveis, Design de Interação

Número de páginas: 175

RESUMO

Depurar algoritmos é uma prática presente no aprendizado de programação. A identificação e a correção de erros possibilitam que o aluno compreenda o problema e a sua própria forma de pensar. Por isso, a depuração de algoritmos esteve entre os princípios de design do LOGO, um dos primeiros ambientes desenvolvidos para crianças aprenderem programar. Brinquedos programáveis descendem do LOGO e tem o mesmo objetivo, de permitir que crianças tenham os primeiros contatos com algoritmos, mas oferecendo interfaces de programação intuitivas e lúdicas. Essas interfaces se dividem em dois grupos: tangíveis e virtuais. As tangíveis favorecem a colaboração e contato com materiais, e as virtuais proporcionam efeitos gráficos poderosos. Isolados, porém, esses dois tipos de interface não favorecem a depuração. O objetivo deste trabalho é explorar como a união de interfaces tangíveis e virtuais pode facilitar a depuração de algoritmos por crianças. Para isso apresenta a RoPE AR, uma interface de realidade aumentada (RA) para o brinquedo programável RoPE. A proposta da RoPE AR é usar a câmera de um smartphone para captar algoritmos criados com blocos de papelão; enviar esses algoritmos para o RoPE executar; e controlar um projetor que emite elementos virtuais sobre os blocos e um tapete projetado. Um mapeamento de 56 brinquedos programáveis identificou que a abordagem sugerida nesta pesquisa não é utilizada, pois apenas 2 brinquedos usam RA e nenhum aplica projeção. A avaliação da RoPE AR ocorreu em um centro de desenvolvimento infantil com a participação de 20 crianças, que interagiram em duplas e trios criando e depurando algoritmos para o RoPE capturar uma maçã projetada. As interações foram filmadas e as professoras foram entrevistadas após as atividades. A interface se mostrou viável para ambientes de sala de aula, e as crianças conseguiram criar, editar, e corrigir erros em algoritmos. Entretanto, não é possível afirmar que o uso da realidade aumentada favoreceu a depuração. Ainda assim, o experimento revela que as crianças perceberam a realidade aumentada, mas não distinguiram claramente os elementos tangíveis e objetos virtuais projetados.

Além disso, o feedback sonoro e visual emitido quando o brinquedo “colide” no objeto virtual parece estimular a conclusão da tarefa. Nas entrevistas, os professores sugeriram que pesquisas futuras possam apoiar a configuração de tapetes virtuais para permitir a exploração de temas abertos e de contação de histórias, indo além de desafios de programação com objetivos pré-definidos.

ROPE AR: AN AUGMENTED REALITY INTERFACE FOR DEBUGGING ALGORITHMS BY CHILDREN 4-6 YEARS OLD

Cesar Pereira Viana

September / 2021

Advisor: André Luis Alice Raabe, Dr.

Area of Concentration: Applied Computer Science

Research Line: Informatics applied to computer teaching

Keywords: Debugging, augmented reality, programmable toys, interaction design

Number of pages: 175

ABSTRACT

Debugging algorithms is a practice present in learning to program. The identification and correction of errors allows the student to understand the problem and their own way of thinking. For this reason, debugging algorithms was among the design principles of LOGO, one of the first environments developed for children to learn to program. Programmable toys descend from LOGO and have the same aim of allowing children to have their first contact with algorithms, but offering intuitive and playful programming interfaces. We divided these interfaces into two groups: tangible and virtual. Tangibles favor collaboration and contact with materials, and virtual ones provide powerful graphic effects. Isolated, however, these two types of interfaces are not conducive to debugging. This work aims to explore how the union of tangible and virtual interfaces can facilitate the debugging of algorithms by children. For that, it presents RoPE AR, an augmented reality (AR) interface for the RoPE programmable toy. RoPE AR's proposal is to use a smartphone camera to capture algorithms created with cardboard blocks; send these algorithms to RoPE to execute; and control a projector that emits virtual elements on the blocks and a projected mat. A mapping of 56 programmable toys identified that the approach suggested in this research is new, as only 2 toys use AR and none apply projection. The RoPE AR evaluation took place at a child development center with the participation of 20 children who interacted in pairs and trios, creating and debugging algorithms for RoPE to capture a projected apple. We filmed the interactions and interviewed the teachers after the activities. The interface proved workable for classroom environments, and children could create, edit, and correct errors in algorithms. However, it is not possible to state that the use of augmented reality favored debugging. Still, the experiment reveals that children perceived augmented reality, but there was no clear distinction between tangible elements and projected virtual objects. In addition, they also perceived the sound and visual feedback emitted when the toy "collides" with the virtual object. In the interviews, the teachers suggested future research could support the configuration of virtual mats to allow to explore open subjects and storytelling, going further than programming challenges with pre-defined objectives.

LISTA DE FIGURAS

Figura 1.	Espiral de aprendizagem.....	18
Figura 2.	Brinquedos com interfaces tangíveis.	20
Figura 3.	Brinquedo RoPE.....	21
Figura 4.	Solução proposta: interface tangível com realidade aumentada.....	23
Figura 5.	Processo de depuração proposto por Carver e Klahr (1986) no contexto de um BP. .	39
Figura 6.	Modelos de depuração.	40
Figura 7.	Classificação de brinquedos programáveis por características físicas.	44
Figura 8.	Visão superior e frontal do RoPE.....	47
Figura 9.	Aplicativo para programar o RoPE.	48
Figura 10.	Tapetes temáticos.....	49
Figura 11.	Blocos do Scratch e do ScratchJr.	51
Figura 12.	Blocos tangíveis.	52
Figura 13.	Exemplos de marcas fiduciais.	53
Figura 14.	Exemplos de uso da biblioteca TopCodes.	54
Figura 15.	<i>Continuum</i> de realidade-virtualidade de Milgram (1994).	55
Figura 16.	Interseção de objetos virtuais entre o olho humano e o mundo real.	56
Figura 17.	Dispositivos para criar realidade aumentada.....	58
Figura 18.	Aplicações de RA Projetiva.....	60
Figura 19.	Ferramentas educacionais usando RA projetiva.	61
Figura 20.	Robo-Blocks.....	65
Figura 21.	Cubetto: indicadores luminosos, área de função e blocos.	67
Figura 22.	Mesa interativa com a Bee-Bot.	69
Figura 23.	ALERT.....	70
Figura 24.	Brinquedos lançados por ano entre 2003 e 2019.....	82
Figura 25.	Interesse pelo tema Pensamento Computacional, segundo o Google Trends.	82
Figura 26.	Método Kawakita Jiro.....	83
Figura 27.	Agrupamento das interfaces.	84
Figura 28.	Brinquedos por tipo de interface.....	85
Figura 29.	Usos de RA com brinquedos programáveis.....	86
Figura 30.	Conceitos de programação abordados por brinquedos programáveis.....	87
Figura 31.	RoPE AR em uso. Círculo projetado mostra bloco que forma o algoritmo.	89
Figura 32.	Telas do aplicativo.	90
Figura 33.	Elementos tangíveis. 1 - RoPE. 2 - Blocos de código. 3 - Marcadores de calibragem. 4 - Ativador de depuração.	92
Figura 34.	Visão geral dos componentes do sistema.	92
Figura 35.	Comparativo de dimensões de foto e vídeo para os smartphones utilizados.	94
Figura 36.	Aplicação do padrão Composite.....	96
Figura 37.	Conexão entre o módulo HM-10 e a placa do brinquedo RoPE.	99
Figura 38.	Diagrama de classes do firmware.....	100
Figura 39.	Dados do Projeto Político Pedagógico do CDI (2020).	105
Figura 41.	Distribuição de idades.....	107
Figura 42.	Posicionamento do projetor.	108

Figura 43.	Especificação da mecânica.....	114
Figura 44.	Item configurado na plataforma e instanciado no aplicativo RoPE AR.	115
Figura 45.	Processo de categorização.....	116
Figura 46.	Marcações de trechos dos vídeos em cores que indicam seu código.....	117
Figura 48.	Exploração dos blocos	124
Figura 49.	Percentual de ações em cada estado para todas as equipes, somente nas fases de programação.....	128
Figura 50.	Percentual de ações em cada estado para todas as equipes, nas fases de depuração ..	128
Figura 51.	Distribuição de tempo por fases.	129

LISTA DE TABELAS

Tabela 2.	Resultados da RSL.....	74
-----------	------------------------	----

LISTA DE QUADROS

Quadro 1.	Pilares do Pensamento Computacional.	36
Quadro 2.	Como brinquedos programáveis e kits promovem conceitos e práticas computacionais.	46
Quadro 3.	Conceitos e práticas computacionais e sua promoção pelo brinquedo RoPE.....	48
Quadro 4.	Análise comparativa.	72
Quadro 5.	Critérios de inclusão e exclusão.	73
Quadro 6.	Artigos resultantes da RSL	75
Quadro 7.	Brinquedos resultantes do mapeamento industrial.	81
Quadro 8.	Ligação entre código nativo escrito e código Kotlin.	95
Quadro 9.	Implementação da classe de composição.....	97
Quadro 10.	Protocolo de comunicação.....	101
Quadro 11.	Ligação entre caracteres e funções executadas no brinquedo.	101
Quadro 12.	Encontros e participantes	107
Quadro 13.	Fases de programação	112
Quadro 14.	Fases de depuração	113
Quadro 15.	Questionário com as professoras.....	113
Quadro 16.	Equipes.	118
Quadro 17.	Falas sobre os símbolos dos blocos.....	120
Quadro 18.	Reconhecimento dos blocos.....	121
Quadro 19.	Eventos de percepção de realidade aumentada	124
Quadro 20.	Ações mapeadas.....	126
Quadro 21.	Estados	127

LISTA DE ABREVIATURAS

BPs Brinquedos Programáveis

BP brinquedo programável

CDI Centro de Desenvolvimento Infantil

CSTA *Computer Science Teachers Association*

IBGE Instituto Brasileiro de Geografia e Estatística

IDEB Índice de Desenvolvimento da Educação Básica

IHC Interação Humano Computador

ISTE *International Society for Technology in Education*

PC Pensamento Computacional

PPP Projeto Político Pedagógico

CAQDAS *Computer Assisted Qualitative Data Analysis Software* - Software de Apoio à Análise de Dados Qualitativos

RA Realidade Aumentada

RAE Realidade Aumentada Espacial

RAEP Realidade Aumentada Espacial Projetiva

SUMÁRIO

1	INTRODUÇÃO	17
1.1	PROBLEMA DE PESQUISA	21
1.1.1	Solução Proposta	23
1.1.2	Delimitação de Escopo.....	24
1.1.3	Justificativa	24
1.2	OBJETIVOS	26
1.2.1	Objetivo Geral	26
1.2.2	Objetivos Específicos.....	26
1.3	METODOLOGIA	26
1.3.1	Métodos da Pesquisa	26
1.3.2	Procedimentos Metodológicos	27
1.4	ESTRUTURA DA DISSERTAÇÃO	27
2	FUNDAMENTAÇÃO TEÓRICA	29
2.1	A CRIANÇA	29
2.1.1	Piaget e o Desenvolvimento Cognitivo.....	30
2.1.2	Jerome Bruner: Construtivismo no Ensino.....	31
2.1.3	Papert e o Construcionismo	32
2.2	PENSAMENTO COMPUTACIONAL	34
2.2.1	Depuração.....	37
2.2.2	Depuração e visibilidade.....	39
2.3	INTERFACES DE PROGRAMAÇÃO PARA CRIANÇAS	42
2.3.1	Brinquedos Programáveis.....	43
2.3.2	Interfaces Tangíveis	49
2.3.3	Programação em Blocos	51
2.3.4	Marcas Fiduciais	52
2.4	REALIDADE AUMENTADA	53
2.4.1	Tecnologias para Criar Realidade Aumentada	55
2.4.2	RA Projetiva.....	59
2.5	CONSIDERAÇÕES	62
3	TRABALHOS RELACIONADOS	64
3.1	INTERFACES DE BRINQUEDOS PROGRAMÁVEIS	64
3.1.1	Robo-Blocks	65
3.1.2	Cubetto	66
3.1.3	Bee-Bot com mesa interativa	68
3.1.4	ALERT	69
3.1.5	Análise Comparativa	70
3.2	REVISÃO SISTEMÁTICA: PESQUISAS DE IHC COM CRIANÇAS	72
3.2.1	Resultados	73

3.2.2	QE1 — Quais categorias de interfaces são usadas em pesquisas sobre aprendizado de algoritmos na Educação Infantil?	77
3.2.3	QE2 — Quais métodos e design de experimentos são utilizados?	77
3.2.4	QE3 — O que as crianças aprendem ao interagir com brinquedos programáveis?	78
3.3	MAPEAMENTO INDUSTRIAL: INTERFACES DE BRINQUEDOS PROGRAMÁVEIS	78
3.3.1	Questões de Pesquisa	79
3.3.2	Processo de Busca	79
3.3.3	Filtro	80
3.3.4	Resultados	80
3.3.5	QE1 - Quais são as categorias de interfaces mais frequentemente utilizados em brinquedos programáveis?	82
3.3.6	QE2 - Como a realidade aumentada é aplicada em brinquedos programáveis?	86
3.3.7	QE3 - Quais são os conceitos de algoritmos abordados nas interfaces de brinquedos programáveis?	86
3.4	CONSIDERAÇÕES	87
4	DESENVOLVIMENTO	89
4.1	ELEMENTOS DA INTERFACE	89
4.1.1	Aplicativo	89
4.1.2	Elementos tangíveis	90
4.2	ASPECTOS TÉCNICOS	91
4.2.1	Smartphone	93
4.2.2	RoPE: Comunicação Bluetooth	98
4.3	CONSIDERAÇÕES	100
5	AVALIAÇÃO	103
5.1	CONTEXTO	103
5.1.1	Município	103
5.1.2	Centro de Desenvolvimento Infantil	104
5.1.3	Proposta Pedagógica para a Educação Infantil	105
5.2	PARTICIPANTES DA PESQUISA	106
5.3	AMBIENTE	108
5.4	INSTRUMENTOS DE COLETA	109
5.4.1	Vídeos de interações	109
5.4.2	Entrevistas com as professoras	112
5.4.3	Registro de respostas na CtPuzzle Platform	114
5.4.4	Métodos de análise dos dados	115
5.5	CONSIDERAÇÕES	117
6	RESULTADOS	118
6.1	ANÁLISE INDUTIVA	118
6.1.1	Interação com os blocos	119
6.1.2	Percepções de realidade aumentada	123

6.2	ANÁLISE DEDUTIVA	125
6.2.1	Tempos por fase	129
6.3	ENTREVISTAS.....	130
7	CONCLUSÃO	134
7.1	CONTRIBUIÇÕES DA DISSERTAÇÃO	137
7.2	TRABALHOS FUTUROS	138
7.3	ASPECTOS ÉTICOS.....	139
	Referências	140
	APÊNDICE A – STRINGS DE BUSCA PARA REVISÃO SIS- TEMÁTICA DA LITERATURA	148
	APÊNDICE B – REQUISITOS DO APLICATIVO.....	149
	APÊNDICE C – MÓDULOS DO APLICATIVO	153
	APÊNDICE D – PROTOTIPAÇÃO.....	155
	APÊNDICE E – MINI-PROJETOR	157
	APÊNDICE F – ENTREVISTAS COM AS PROFESSORAS .	158

1 INTRODUÇÃO

A depuração é entendida como um dos conceitos centrais do aprendizado de programação (CARVER; KLAHR, 1986). Ela compreende a etapa de encontrar e solucionar problemas ou *bugs*¹ em algoritmos. McCauley et al. (2008) menciona que a depuração é um tema de difícil aprendizado para os estudantes e um desafio de ensino para os professores. Neste sentido, a literatura sobre o tema busca responder questões como: quais as causas dos *bugs*, quais categorias de *bugs* são mais frequentes?, e como melhorar o ensino e aprendizado de depuração?

Enquanto no contexto comercial os erros em *software* são indesejados e causam prejuízos de toda ordem (VALDIVIA-GARCIA, 2016), no campo educacional eles representam oportunidades de aprendizado. Segundo Papert (1980), não se espera que qualquer coisa funcione na primeira tentativa, portanto errar e depurar faz parte do processo.

Valente (2018) expressa essa ideia como uma espiral de *descrição-execução-reflexão-depuração* (Figura 1). Ao entrar nessa espiral, primeiramente o aluno descreve um programa e o executa, para obter um resultado. Pode, então, refletir sobre se a execução do seu programa resultou naquilo que intencionava. Por fim, em caso negativo, deve depurar o programa. Essa depuração pode eliminar erros de sintaxe, revelar incompreensão de conceitos envolvidos no problema ou esclarecer quais passos devem ser aplicados para resolvê-lo. Esse ciclo possibilita ao aluno passar de um nível de conhecimento inicial para outro mais elaborado.

A depuração também foi uma das diretrizes de design do ambiente de programação Logo (SOLOMON et al., 2020). Desenvolvido a partir de 1966, ele foi o primeiro ambiente de programação projetado para crianças. Era composto por um computador, uma linguagem textual de programação e um robô com rodas que se movia no chão e desenhava sobre papel. Esse ambiente permitia depurar por meio instruções da própria linguagem, como *pause*, impressão de pilha de comandos e impressão de variáveis.

Além desses recursos técnicos, para Solomon et al. (2020) a "grande ideia" do Logo era permitir à criança depurar seu próprio entendimento do processo computacional e do algoritmo a ser implementado. Um exemplo é o entendimento dos comandos *left* e *right*, que as crianças confundiam com mover para o lado. Para auxiliar a criança depurar esse conceito, um professor poderia sugeri-la

¹ Termo em inglês que, quando ligado à informática, significa falha ou defeito em código que provoca mau funcionamento.

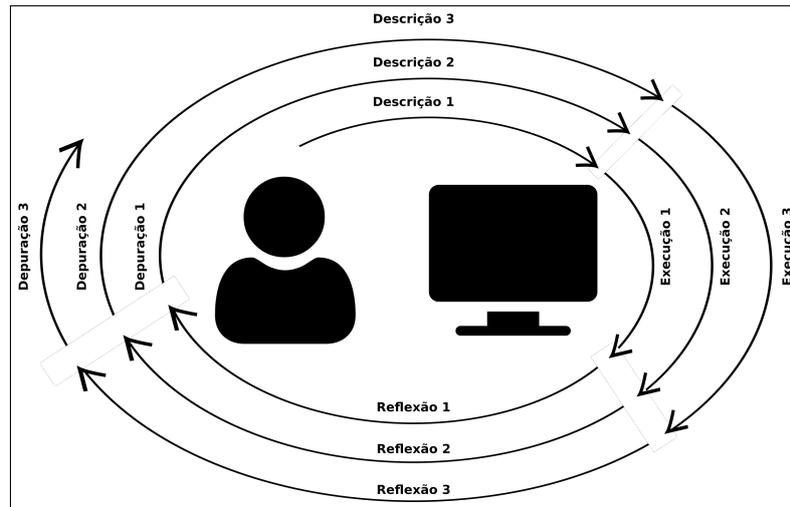


Figura 1. Espiral de aprendizagem.
 Fonte: Adaptado de Valente (2018).

"brincar de ser programada", para então compreender que esses comandos representam movimentos de giro.

A ideia de que a depuração poderia representar uma atividade benéfica para o desenvolvimento cognitivo, somada à existência da linguagem Logo, estimulou pesquisas sobre a prática de depuração por crianças. Carver e Klahr (1986) avaliaram a depuração de algoritmos por crianças de 7 a 8 anos durante um curso usando Logo. Para isso definiram um modelo de depuração em quatro fases: (i) identificação da diferença entre resultado esperado e resultado obtido; (ii) levantamento de hipóteses para a causa do erro; (iii) localização do *bug* no código; e (iv) correção do *bug*. Com base neste modelo, os pesquisadores concluíram que, após 24 horas de curso, as crianças não desenvolveram estratégias efetivas de depuração, preferindo apagar todo o código e escrever novamente. Posteriormente os mesmos autores conseguiram identificar que crianças aprenderam procurar erros e transferiram esta habilidade para atividades não relacionadas à programação (CARVER; RISINGER, 1987).

Além de possibilitar pesquisas sobre como as crianças programam, o ambiente Logo contribuiu para o surgimento, na década de 1990, de um ramo de estudos de Interação Humano Computador (IHC) focado no design de interação para crianças (HOURCADE, 2015). Esse ramo de estudos permitiu identificar dificuldades que as crianças tinham ao usar teclado de computadores convencionais (MCNERNEY, 2004), projetados para adultos. O uso dos teclados exigia alfabetização, coordenação motora fina, e propiciava a ocorrência de erros sintáticos. Esses aspectos desviavam a atenção que deveria estar focada na construção de algoritmos e entendimento do problema para aspectos secundários, como entender o funcionamento da interface.

O reconhecimento dessa inadequação na interação das crianças com o ambiente de programação, motivou a criação de uma classe de ferramentas dos Brinquedos Programáveis (BPs). Os BPs descendem do ambiente Logo e buscam facilitar o aprendizado de programação pelo público infantil. Assim como o robô do ambiente Logo, eles geralmente se movem sobre o chão, e tem aparências que remetem ao imaginário infantil, como animais, carros ou robôs (RAABE et al., 2017). Em geral, sua principal funcionalidade é possibilitar à criança inserir comandos, que o brinquedo então executa. Esses brinquedos têm interfaces de programação projetadas para facilitar à criança concentrar-se nos conceitos fundamentais, sem se distrair com erros que não contribuem para o aprendizado de algoritmos.

As interfaces dos BPs são simplificadas, para que crianças pequenas saibam como interagir. Adicionar controles de depuração nessas interfaces aumenta a sua complexidade. Ambientes de desenvolvimento profissionais e/ou voltados para o ensino de programação introdutória para adultos apresentam controles de depuração. Esses ambientes mostram valores de variáveis, permitem executar passo a passo e definir pontos de parada (NOSCHANG et al., 2014). Se adultos precisam destas ferramentas, será que as crianças poderiam se beneficiar das mesmas funcionalidades caso aplicadas em interfaces de BPs?

Sipitakiat e Nusén (2012) tentam responder essa pergunta ao criar o Robo-Blocks, uma interface tangível de programação em blocos que permite execuções passo a passo. Os autores perceberam que as crianças implementavam “grosseiramente” suas ideias para ver o robô executá-las, sem refletir e aprender com os erros. Além disso, os movimentos rápidos do robô não permitiam comparar as ações do brinquedo com os blocos de “código”, dificultando associar um movimento errado com um bloco. Neste sentido, a execução passo a passo permitiu associar blocos com movimentos, o que foi auxiliado por um LED ativado em cada bloco durante sua execução.

O Cubetto (ANZOATEGUI; PEREIRA; JARRIN, 2017) (Figura 2a) é um brinquedo programável (BP) é programado por um painel onde são encaixados blocos de madeira. O painel destaca cada bloco executado acendendo LEDs. Permite, portanto, que a criança associe o movimento do brinquedo com sua causa na lista de comandos. Diferente do Robo-Blocks, porém, não há a opção de execução passo a passo. O Cubetto e o Robo-Blocks, portanto, tentam atacar o problema da depuração em interfaces que devem ser compreendidos por crianças.

Uma característica comum nestes trabalhos é a presença de componentes eletrônicos. Eles têm baterias, conectores e placas eletrônicas. Como alternativa, Horn e Bers (2019) apresentam as “linguagens externamente compiladas”. Elas são interfaces com blocos tangíveis que contém marcas

especiais, denominadas marcas fiduciais. Essas marcas são lidas por um sensor óptico e interpretadas por um software que identifica as marcas e as converte em um conjunto de comandos para o brinquedo executar. Deste modo os blocos não carecem de componentes eletrônicos, aumentando a liberdade para os projetistas escolherem os materiais ou objetos a utilizar Horn e Bers (2019). O KIBO (SULLIVAN; ELKIN; BERS, 2015) (Figura 2b) é um executado de BP programado por uma linguagem externamente compilada.



(a) Cubetto



(b) KIBO

Figura 2. Brinquedos com interfaces tangíveis.

Por outro lado, sem componentes eletrônicos, como LEDs, os blocos tangíveis carecem de meios para representar as instruções do algoritmo que durante a sua execução. A ausência de um indicador dificulta para a criança relacionar o comando está sendo executado e seus efeitos no comportamento do brinquedo. Portanto, também dificulta encontrar causas de erros, bem como mapear símbolos (imagens presente nos blocos) ações (movimentos do brinquedo).

Uma abordagem possível para diminuir a necessidade de eletrônicos também possibilitar indicadores luminosos na interface de programação é Realidade Aumentada (RA) projetiva (ROBERTO et al., 2013). Ela é uma técnica de projeção de elementos virtuais associados a objetos reais. Ela permite que um projetor crie animações, luzes, ou qualquer objeto graficamente representável e os projete interagindo com objetos reais em uma cena. Permite, portanto, avançar a representação oferecida por LEDs fixos em blocos e painéis, ao variar as cores e formas projetadas. Deste modo, os blocos tangíveis do algoritmo podem ser construídos com materiais diversos e ainda assim proporcionar *feedbacks imediatos* (NORMAN, 1990) durante a depuração.

O RoPE (Robô Programável Educacional) representa uma oportunidade para testar a viabilidade desta abordagem de criação de interfaces tangíveis externamente compiladas e com uso de RA projetiva. Ele é um BP desenvolvido na Univali pelo Laboratório Lite² com foco em crianças a partir dos 3 anos. A sua interface de programação são botões coloridos (Figura 3), desenhados para serem simples de usar por crianças pequenas (RAABE et al., 2017). Essa interface se mostrou um sucesso, e o brinquedo já foi usado em Centros de Desenvolvimento Infantil por mais de 1000 crianças. Além dos botões, pesquisas desenvolveram protótipos de outros modos de interação, como um aplicativo de celular (VIANA, 2018) e uma interface de programação em similar ao Cubetto (METZGER, 2018).



Figura 3. Brinquedo RoPE.

Este trabalho ocorre como uma continuidade dessas e busca contribuir com a área de Informática na Educação ao (1) projetar uma interface tangível externamente compilada para o brinquedo RoPE e (2) explorar o uso de RA projetiva para facilitar a depuração. Essa contribuição se fundamenta na literatura sobre interfaces tangíveis para crianças (SAPOUNIDIS; DEMETRIADIS; STAMELOS, 2015; HORN; BERS, 2019; PLOWMAN; LUCKIN, 2004) e na observação de que o uso de RA projetiva é um campo pouco explorado no contexto de BPs.

1.1 PROBLEMA DE PESQUISA

O problema que este trabalho se propõe a investigar no contexto do brinquedo RoPE se insere no que Norman (1990) denomina falta de *visibilidade*. A visibilidade é um princípio de design relacionado a como o usuário percebe estado de um sistema, e como ele mapeia as ações que pode fazer e que alterações de estado elas provocam.

² <<https://lite.acad.univali.br>>

Em BPs que possuem apenas botões, como a Bee-Bot, o algoritmo que a criança programa não é visível. A criança não vê a sequência de comandos construída ao apertar os botões. Essa invisibilidade dificulta conversar sobre o algoritmo, analisá-lo, encontrar erros e desfazer operações. Raabe et al. (2015) menciona esse a incompreensão do funcionamento do botão de limpar a memória da Bee-Bot. As crianças esqueciam de limpar os comandos programados anteriormente, dado que essa ação não é automática. O estado do sistema não é visível, e a criança não sabe se os comandos foram limpos. A solução sugerida foi adicionar um visor de comandos.

No caso das interfaces de blocos tangíveis externamente compiladas (como a utilizada pelo KIBO), o problema é a *invisibilidade de execução* do algoritmo. Os blocos carecem de indicadores que mostrem qual bloco está sendo executado em cada momento, correspondente a cada movimento do brinquedo. Essa falta de visibilidade de execução dificulta compreender a sequência lógica de funcionamento do algoritmo, pois não há mapeamento entre comando/bloco e ação do robô. O Cubetto soluciona esse problema através de luzes indicadoras, e para isso depende de um painel onde os blocos são encaixados.

Norman (1990) comenta que dispositivos cujas interfaces não demonstram o estado interno durante a interação podem influenciar o entendimento do usuário sobre como tal dispositivo funciona. Esse “entendimento” é denominado modelo mental. Caso o dispositivo executar de forma inesperada, o usuário tende a encontrar alguma explicação para justificar tal comportamento. Ao justificá-lo, cria um modelo mental incorreto. Raabe et al. (2015) apresentam crianças justificando comportamentos inesperados da Bee-Bot após terem esquecido de limpar sua memória, afirmando que a “abelha” tinha vontade própria.

Por fim, essa falta de visibilidade desencadeia a dificuldade em depurar. A literatura (CARVER; KLAHR, 1986; MCCAULEY et al., 2008) apresenta modelos organizam a ação de depurar em etapas, entre as quais está a de localizar e reparação erros em algum código-fonte. A ausência do código-fonte, portanto, impede a depuração segundo estes modelos.

Partindo destas reflexões, elaborou-se a seguinte pergunta de pesquisa: *como crianças de 4 a 6 anos interagem com uma interface de programação focada em permitir visualizar os passos e a execução de um algoritmo?*

1.1.1 Solução Proposta

A proposta é desenvolver uma interface de RA projetiva combinada com uma linguagem externamente compilada. Dado que os blocos de linguagens externamente compiladas tem marcas fiduciais, uma câmera pode captar as posições dos blocos e um algoritmo pode controlar um projetor para destacá-los Figura 4.

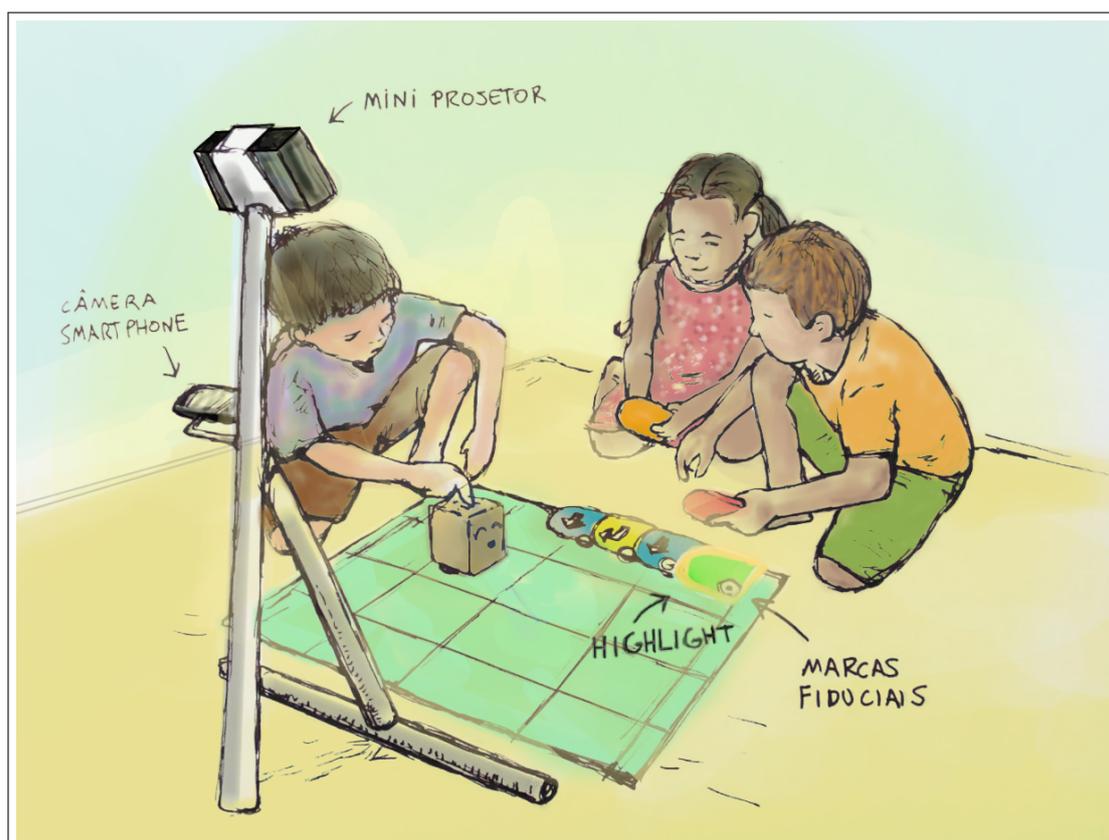


Figura 4. Solução proposta: interface tangível com realidade aumentada.

Fonte: O autor.

A intenção é que a criança programe utilizando blocos de papelão com símbolos de setas correspondentes às ações do RoPE. Um *smartphone* é usado como câmera, pois é mais acessível e fácil de manusear do que um notebook ou desktop. Um projetor portátil, de baixa potência de iluminação (1200 lumens) e de baixo custo ilumina os blocos e projeta um mapa interativo.

Outra funcionalidade implementada é o estado de *execução passo a passo*. Ao ativar esse estado o brinquedo executa um comando e espera a criança disparar a execução da próxima ação. Isso possibilita à criança mais tempo para olhar cada bloco e compreender o movimento associado a ele. A intenção é que a criança possa compreender possíveis erros no algoritmo construído por ela ou por outra criança.

A partir do uso de RA e de um estado de execução passo a passo, a proposta é possibilitar um melhor entendimento, por parte da criança, a respeito do algoritmo que ela está construindo, e sobre seu próprio modo de pensar. Esta abordagem é exploratória, e não foram encontrados dados quantitativos passíveis de comparações numéricas com outros estudos. Este trabalho, portanto, apresenta a construção da proposta e a avaliação qualitativa das interações de crianças de 4 a 6 anos com a mesma.

1.1.2 Delimitação de Escopo

Este trabalho é sobre uma ferramenta para facilitar o aprendizado de algoritmos por crianças. Algoritmos é um dos quatro pilares do Pensamento Computacional (BRACKMANN, 2017), junto à decomposição, abstração e reconhecimento de padrões. Este trabalho cita definições destes pilares, porém a solução proposta tem foco apenas em algoritmos, e, mais especificamente, na atividade de depurá-los.

Não é objetivo deste trabalho realizar testes com comprovação estatística, no sentido de identificar relações de causa e efeito no aprendizado. O foco é descrever a presença ou ausência das dificuldades mais perceptíveis durante a interação com uma ferramenta educacional. Esse foco se adequa à pesquisa de cunho qualitativo, com um número reduzido de crianças.

Também não foi objetivo programar um algoritmo de visão computacional para reconhecer os blocos tangíveis e o RoPE. Uma biblioteca de identificação de marcas fiduciais permitiu agilizar a implementação de um protótipo funcional. A eliminação de marcas fiduciais poderá ocorrer em trabalhos futuros, considerando os aprendizados obtidos com a primeira versão aqui apresentada.

1.1.3 Justificativa

O mapeamento industrial descrito no Capítulo 3, demonstra que, de 56 brinquedos, apenas 2 utilizam RA. Ainda assim o seu uso ocorre através de tablets e celulares, menos indicados para crianças em comparação com as interfaces tangíveis (SAPOUNIDIS et al., 2019; ZUCKERMAN; GAL-OZ, 2013).

No mesmo mapeamento, o RoPE aparece como o único projeto brasileiro em atividade voltado ao ensino de programação para crianças pequenas com brinquedos programáveis. Desde a sua concepção, o projeto estuda aspectos de design para que crianças possam aprender de forma lúdica e evitando erros não construtivos (RAABE et al., 2015). A abordagem aplicando RA, como este

trabalho propõe, ainda não foi experimentado com o RoPE ou outros brinquedos. Considerando o pioneirismo do RoPE como ferramenta educacional para aprendizado de algoritmos por crianças no Brasil, entende-se que a exploração proposta neste trabalho pode contribuir com novas informações para o projeto.

A reprodução da proposta é facilitada pelo fato da mesma não utilizar componentes eletrônicos específicos. Os componentes usados são produtos disponíveis comercialmente, que precisam ser plugados (mini-projetores, *smartphones* e Arduino). O RoPE compartilha princípio de ser programado por botões com diversos outros brinquedos, como a Bee-Bot, Go Robot Mouse e Thymio. Esses brinquedos também poderiam ser aplicados no contexto da proposta, o que também facilita sua reprodução.

A proposta de usar RA projetiva abre caminho para desenvolver tapetes dinâmicos. Atualmente o brinquedo RoPE se move sobre tapetes estáticos, que precisam ser trocados conforme a atividade. A projeção, associado à informação da localização do brinquedo, permite disparar eventos quando este atinge determinadas posições. Esses eventos viabilizam a construção de jogos, em que as fases avançam conforme os movimentos do brinquedo. Portanto, ao permitir que um sistema computacional perceba a posição do brinquedo e projete o ambiente, surgem novas possibilidades de interação.

Novas possibilidades de interação também surgem ao utilizar a programação em blocos. Por estarem desassociados dos botões do RoPE, os blocos podem expandir as funções já existentes sem a necessidade de modificações físicas no brinquedo. Enquanto os botões definem ações específicas, como giros de 90.º, adequadas para as crianças iniciarem os primeiros algoritmos, os blocos podem definir outros movimentos, como giros de 45.º, por exemplo. Além disso, outros blocos podem programar eventos de sons e luzes, e também animações como trocar o mapa projetado.

Entretanto, prosseguir com essas possibilidades exige avaliar a ferramenta no em centros de desenvolvimento infantil com crianças de 4 a 6 anos. Essa avaliação é necessária por dois motivos principais. Primeiro, obter a opinião de professores e crianças sobre a interface; e segundo, verificar riscos, como a criança obstruir as marcas fiduciais, ou os elementos físicos e virtuais representarem uma fonte de distração e não de aprendizado.

Por fim, a proposta está ligada a uma área de interesse da comunidade científica³, e ao fato de que há uma lacuna de compreensão sobre utilização de RA projetiva com brinquedos programáveis.

³ Possui jornais dedicados ao tema, como International Journal of Child-Computer Interaction e ACM Transactions on Computer-Human Interaction.

1.2 OBJETIVOS

1.2.1 Objetivo Geral

Avaliar qualitativamente o uso de uma interface de RA projetiva na depuração de algoritmos do brinquedo RoPE por crianças de 4 a 6 anos.

1.2.2 Objetivos Específicos

Os objetivos específicos do trabalho são:

1. Relacionar sistemas e aplicações que utilizam realidade aumentada projetiva;
2. Categorizar as interfaces de brinquedos programáveis existentes;
3. Construir a interface de realidade aumentada projetiva para o brinquedo RoPE;
4. Aplicar a interface em atividades com crianças para coletar e analisar eventos de interação.

1.3 METODOLOGIA

Esta seção descreve a metodologia e os procedimentos metodológicos utilizados nesta pesquisa.

1.3.1 Métodos da Pesquisa

Métodos científicos são conjuntos de procedimentos técnicos adotados para produzir conhecimento (GIL, 2008), construído através de passos para atingir os objetivos de uma pesquisa (WAZLAWICK, 2009). O método desta pesquisa pode ser descrito como de natureza aplicada, pois prevê a aplicação prática de uma ferramenta e tecnologias para solucionar um problema de interação com brinquedos programáveis. A abordagem sobre o problema foi qualitativa, ao observar aspectos subjetivos como falas, gestos e expressões durante o uso da solução proposta. Por fim, esta pesquisa é exploratória, ao buscar extrair informações de um fenômeno desconhecido, a partir de uma pequena amostra de participantes em caráter experimental. A próxima subseção descreve os procedimentos metodológicos adotados.

1.3.2 Procedimentos Metodológicos

A construção da solução proposta e atendimento dos objetivos de pesquisa possui quatro fases: estudo, modelagem, implementação e avaliação, descritas a seguir.

Estudo Essa etapa correspondeu a uma revisão sistemática, um mapeamento industrial e um levantamento de trabalhos similares. A revisão sistemática focou em compreender como são feitas pesquisas científicas avaliando interfaces de programação de brinquedos programáveis. O mapeamento buscou descrever as categorias de interfaces mais utilizados em projetos de brinquedos programáveis, incluindo produtos comerciais. Por último, trabalhos similares descrevem o uso de projeção ou formas de facilitar a programação por crianças.

Modelagem A etapa de modelagem ocorreu como um planejamento dos elementos de software a serem implementados. Esta etapa envolveu levantamento de requisitos (Apêndice B) e prototipação (Apêndice D) para conferir a viabilidade da proposta.

Implementação Etapa que correspondeu a programar o aplicativo (i) alterar o *firmware* do brinquedo RoPE para comunicação sem fio; (ii) configurar a conexão com uma plataforma de armazenamento das interações ocorridas com um brinquedo programável; (ii) adaptar o *firmware* do RoPE para comunicação sem fio; e (iii) programar um aplicativo de *smartphone* para captar os blocos e comunicar com o projetor.

Avaliação A avaliação⁴ aqui compreendida na observação de crianças em atividades de programação do RoPE, para coletar e por fim analisar dados seguindo um protocolo definido no Capítulo 5. Essa etapa envolveu aplicação do software de análise qualitativa Qualcoder, para tratamento dos dados coletados.

1.4 ESTRUTURA DA DISSERTAÇÃO

O trabalho está organizado em 6 capítulos correlacionados. O Capítulo 1, contextualizou e apresentou o tema, o problema e uma proposta de como solucioná-lo. Foram estabelecidos os objetivos a serem alcançados com a solução, e as limitações de escopo a serem observadas.

O Capítulo 2 apresenta a fundamentação teórica iniciando com a descrição do sujeito desta pesquisa (a criança) e observa fundamentos relacionados ao desenvolvimento infantil. Como tema

⁴ Para atendimento do objetivo específico Item 4.

relacionado a algoritmos, é apresentada a definição de Pensamento Computacional (PC) em quatro pilares. O capítulo também apresenta fundamentos sobre as interfaces de brinquedos programáveis, bem como aplicações de realidade aumentada projetiva.

O Capítulo 3 apresenta o estado da arte em três aspectos: das pesquisas realizadas com brinquedos programáveis; interfaces de brinquedos mais utilizados no mercado; e trabalhos com semelhanças tecnológicas e conceituais.

Capítulo 4 é o quarto capítulo, que apresenta uma visão geral dos componentes da contribuição: aplicativo e elementos tangíveis. O capítulo apresenta aspectos técnicos da implementação, tecnologias utilizadas para construir esses componentes e possibilitar a comunicação entre os mesmos.

O Capítulo 5 descreve o contexto de execução da pesquisa. Ele também apresenta o protocolo de teste e os instrumentos de coleta usados durante atividades com crianças. Em seguida, o Capítulo 6 apresenta os resultados obtidos com a análise dos dados coletados.

Por fim, o Capítulo 7 revisa o atendimento aos objetivos da pesquisa, reflete sobre as contribuições geradas e propõe trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo discute os principais temas de interesse desta pesquisa, iniciando pelo sujeito principal: a criança. As bases teóricas são as teorias do Construtivismo e do Construcionismo, e os autores são Jean Piaget, Seymour Papert e Jerome Bruner. A Seção 2.2 apresenta a definição de PC que esta pesquisa segue. Depois o texto apresenta ferramentas voltadas ao aprendizado de algoritmos por crianças, como os brinquedos programáveis.

2.1 A CRIANÇA

No Brasil, o Estatuto da Criança e do Adolescente considera como criança a pessoa de até 12 anos incompletos (BRASIL, 1990), porém Markopoulos et al. (2008) afirma que a duração da infância é relativa a influências culturais, localização e base familiar. O desenvolvimento cognitivo da criança desde o nascimento até a idade adulta foi um dos temas de pesquisa de Piaget, que posteriormente influenciou as pesquisas de Seymour Papert, criador do ambiente Logo.

Compreender como as crianças aprendem é fundamental para projetar ferramentas educacionais efetivas. Essas ferramentas promovem experiências interativas que permitem ao usuário aprender algo. Para Rogers, Sharp e Preece (2013), projetar experiências interativas passa por entender os usuários nos contextos em que vivem, trabalham e também aprendem. Para isso é crucial a observar o indivíduo em seu contexto, para revelar opiniões incorretas, por parte dos designers, sobre o que um grupo deseja ou realmente necessita. Quando o público-alvo do produto interativo é são crianças, é ainda mais difícil para designers, adultos, preverem o que pode ser adequado e simples de usar.

Para abordar a temática do público-alvo desta pesquisa e como ele aprende, esta seção apresenta teorias de aprendizado de Jean Piaget, Jerome Bruner e Seymour Papert. Esses autores buscaram compreender como o conhecimento se constrói na mente da criança, e como materiais externos podem auxiliá-la a aprender melhor. Gray e MacBlain (2015) apresenta uma visão completa de teorias de aprendizagem, mas os autores aqui apresentados se justificam pela influência que tiveram no campo das tecnologias educacionais para crianças.

2.1.1 Piaget e o Desenvolvimento Cognitivo

Piaget estudou o desenvolvimento cognitivo da criança e propôs a teoria denominada Epistemologia Genética. Essa teoria une as correntes filosóficas do apriorismo e do empirismo. Enquanto o apriorismo preconiza que todo conhecimento é inato, ou seja, existe no ser humano desde o nascimento, o empirismo defende que o aprendizado e o conhecimento surgem das interações do sujeito com seu meio. Piaget concorda com o empirismo ao afirmar que o conhecimento surge das experiências e também concorda com o apriorismo quando diz que o aprendizado depende de estruturas mentais capazes de assimilar essas experiências.

A partir disso, Piaget descreveu o aprendizado como um processo de assimilação e acomodação. A criança nasce com poucas estruturas mentais (esquemas), e ao receber estímulos motores, conceituais ou perceptuais, tenta classificá-los e assimilá-los de acordo com aquilo que já conhece. Por exemplo, ao ver um cachorro pela primeira vez a criança pode assimilá-lo como sendo outro animal semelhante para o qual já tenha um esquema criado. Ao ser corrigida por um adulto, porém, um novo conceito surge e precisa ser acomodado, criando assim um novo esquema.

Além de descrever o processo de aprendizado, o pesquisador descreve estágios de desenvolvimento cognitivo. Cada estágio se caracteriza pelo modo como a criança realiza determinadas operações mentais. A realização de uma “operação” significa executar uma ação mental com reversão e conservação. Um exemplo conservação é observar que um líquido transvasado de um recipiente maior para outros dois recipientes menores “conserva” sua quantidade total. No mesmo exemplo, o líquido pode ser “revertido” ao recipiente inicial sem haver mudança de volume. Piaget demonstrou que crianças não conseguem perceber essa relação e sugeriu que isso se devia à ausência das estruturas mentais necessárias para tal. Neste sentido, o pesquisador observou as operações mentais que crianças conseguiam executar em cada idade e com isso organizou o desenvolvimento cognitivo em quatro estágios:

Sensório motor é o primeiro estágio. Inicia no nascimento e vai até os 18 meses (PIAGET, 1964).

Nesta etapa não há conservação ou reversão: quando um objeto some do campo de visão da criança ele deixa de existir para ela. Montessori (2019) indica não mudar frequentemente o ambiente da criança, para ela poder identificar a permanência dos objetos.

Pré-operacional é o estágio que inicia aos 18 meses e vai até os 6 ou 7 anos. Ainda não há estruturas que suportam operações mentais, mas há o aparecimento da linguagem e dos símbolos, o que possibilita o pensar.

Operatório-concreto é o estágio onde se iniciam as operações. Ele é denominado “concreto”, pois as operações dependem de objetos físicos, ou seja, a operação mental precisa estar ligada a uma ação física. Neste período a criança consegue classificar, ordenar objetos e contar.

Operatório-formal é o último estágio, e vai dos 12 anos até o final da vida. Há raciocínio sobre hipóteses abstratas, impossíveis até de serem concretizadas fisicamente. Há suporte para captar informações de fontes variadas e obter conclusões, o que é mais complexo que o raciocínio ligado a ordenação e classificação (PIAGET, 1964).

Percebe-se, a partir da descrição do estágio operatório-concreto, a relação de objetos concretos na realização de operações mentais por crianças entre 7 e 12 anos. Segundo Hourcade (2015), a pesquisa de Piaget sobre como a criança aprende afetou fortemente os campos da IHC. A importância de materiais concretos até hoje é observável em ambientes educacionais infantis. No campo das interfaces de programação pode ser observado nas interfaces tangíveis, que permitem à criança programar utilizando objetos físicos.

2.1.2 Jerome Bruner: Construtivismo no Ensino

Assim como Piaget, Bruner foi um psicólogo construtivista e apoiava a ideia de que o conhecimento é construído pelo indivíduo ao viver diferentes experiências. Porém, enquanto Piaget se concentrou na aprendizagem, Bruner estudou no ensino e mais especificamente em como melhorá-lo. Neste sentido, Bruner destaca três aspectos para melhorar o ensino: representação do aprendizado, currículo em espiral e aprendizagem por descobertas.

A representação do aprendizado seria a forma de adquirir e internalizar o conhecimento, e também ocorre em estágios. Há três modos de representação: enativa, icônica e simbólica. A primeira aconteceria com experiências concretas, “mão na massa”, com estímulos sensoriais tangíveis. Um exemplo seria a criança dividir uma laranja em duas partes. Na segunda fase — representação icônica — o aprendiz associa as experiências sensoriais anteriores com imagens e figuras semelhantes com aquilo que representam. A figura de uma laranja dividida em partes é associada com a experiência concreta de ver uma laranja real partida em pedaços.

Esse espectro de representação do concreto/sensorial para o abstrato/formal pode traçar um paralelo com as interfaces de programação. A criança precisa ter contato físico, enativo com algo. Esse contato facilita a compreensão do algoritmo criados com ícones representando objetos reais,

como usados em programação em blocos virtuais (FLANNERY et al., 2013). Por fim, o contato com programas criados com ícones permite entender a lógica de programação e alcançar o uso de linguagens textuais.

O segundo aspecto que Bruner defende é o currículo em espiral. A ideia é que cada atividade de ensino deve repetir fundamentos lecionados em uma etapa anterior e adicionar novas camadas de complexidade. À medida que a complexidade aumenta, o aluno tem necessidade de um auxílio para entender determinados conceitos. Aí entra o papel de um tutor com a informação a ser transmitida, proporcionando o que Bruner denomina *scaffolding*. No *scaffolding* o tutor auxilia o estudante no aprendizado inicial de um conteúdo (VALKENBURG, 2010) e remove a assistência à medida que o aprendiz adquire autonomia. Para isso o tutor *scaffolding* avalia o que o estudante já sabe e trabalha somente as suas dificuldades, garantindo que ele faça a maior parte do trabalho de forma independente (VALKENBURG, 2010).

Projetar ferramentas que incentivem o *scaffolding* entre adultos e crianças é uma preocupação do campo de design de interação para crianças. Plowman e Luckin (2004) observam que a presença de um elemento tangível pode aumentar o número de vezes em que adultos auxiliam crianças. Horn, Crouser e Bers (2012) afirmam que as interfaces tangíveis combinadas com as interfaces gráficas tem o potencial de fornecer *scaffolding* à medida que os estudantes podem transitar de um sistema tangível para sistemas gráficos com recursos e complexidades aumentados. Já Catlin et al. (2018) comenta o mesmo sobre a robótica educacional:

Robots allow teachers to create environments which reflect Bruner's Spiral Curriculum [...]. Young children start with [a robot] where all they do is put symbols in the right order, but as their experience and interest grows they can end up coding in professional programming languages. Several robots provide rich educational environments by offering different ways for students to program them (CATLIN et al., 2018).

Por fim, Bruner defende o aprendizado por descobertas. Segundo ele, o aprendiz deve estar motivado pela curiosidade. Representar conceitos adequadamente e auxiliar o aprendiz em suas dificuldades são formas de contribuir para que sua curiosidade se transforme em conhecimento.

2.1.3 Papert e o Construcionismo

Seymour Papert também estudou o desenvolvimento infantil. Enquanto Piaget e Bruner são construtivistas, Papert criou a teoria denominada Construcionismo. Ela tem base no construtivismo de

Piaget, com quem Papert trabalhou, e ambas as teorias compartilham a ideia de “construir estruturas de conhecimento” (PAPERT; HAREL, 1991). A diferença é que Construtivismo se concentra em como as estruturas são construídas na mente do aprendiz, e o Construcionismo foca em como materiais externos podem auxiliar na construção dessas estruturas de conhecimento (BERS, 2008).

O foco nos materiais e ferramentas decorre da crença de que objetos bem projetados possibilitam criar projetos significativos do ponto de vista epistemológico (BERS, 2008). O material deve ter uma finalidade aberta, permitindo criar em vez de apenas usar algo pronto. Exemplos desses materiais são blocos de encaixar, o computador e a robótica. Eles permitem combinar diferentes partes, blocos e estruturas para testar hipóteses, errar e avaliar possibilidades. Essa versatilidade possibilita ao aprendiz criar artefatos do seu interesse, o que aumenta a motivação e a curiosidade assim como a aprendizagem por descobertas proposta por Bruner. Papert (1972) comentava:

Eu acredito com Dewey, Montessori e Piaget que as crianças aprendem fazendo e pensando sobre o que fazem. Portanto, os ingredientes fundamentais da inovação educacional devem ser coisas melhores para fazer e maneiras melhores de pensar sobre si mesmo fazendo essas coisas (PAPERT, 1972, p.3, tradução nossa).

A liberdade da programação está em criar blocos de código e combiná-los de infinitas maneiras e gerar infinitos resultados. Percebendo essa harmonia com o Construcionismo, Papert e seu grupo de pesquisa desenvolveram a linguagem LOGO, a primeira linguagem de programação criada para e por crianças (SOLOMON et al., 2020). A linguagem LOGO é conhecida como a “linguagem da tartaruga” ao permitir programar um agente para se mover, girar e desenhar. Essa tartaruga pode ser digital, na tela de um computador; ou física, na forma de brinquedos programáveis.

Desde o desenvolvimento do LOGO, pesquisas têm se inspirado nos fundamentos construcionistas para criar ferramentas adaptadas para crianças. Exemplos são ambientes de programação em blocos como o Scratch e o ScratchJr (FLANNERY et al., 2013), e diversos brinquedos programáveis (ver Subseção 2.3.1). Os ambientes de programação em blocos permitem encaixar blocos de código evitando erros de sintaxe comuns na programação da digitação em teclado, como no caso do LOGO. Já os BPs permitem infinitos programas ainda que com número reduzido de comandos disponíveis. Essas ferramentas carregam os princípios construcionistas de serem fáceis de começar, mas infinitas nas possibilidades de criar, testar hipóteses, e compartilhar criações significativas.

Bers (2008) adaptou os princípios construcionistas para o contexto da Educação Infantil, dos quais dois fundamentam esse trabalho. O primeiro deles é “*Usar objetos concretos para construir e*

explorar o mundo". O estágio operatório concreto proposto por Piaget menciona a necessidade que crianças têm de utilizar objetos concretos para realizar operações mentais. Crianças precisam se engajar em atividades com objetos reais, com manipulação de brinquedos que as façam pensar. O modo de interação mais proeminente neste sentido são as interfaces tangíveis. O segundo princípio é "*Engajar em autorreflexão como parte do processo*". Depurar algoritmos representa essa reflexão, ou seja, o momento que a criança pensa sobre aquilo que criou. Ao encontrar um erro, ocorre então o processo de acomodação, ou seja, o resultado não se "encaixa" nos esquemas existentes na mente da criança e precisa ser acomodado. A criança pode então alterar o algoritmo, e ao executar novamente, perceber se o resultado pode ser assimilado pelo esquema mental recém-criado, confirmando ou refutando-o.

As teorias apresentadas nesta seção representam apenas um recorte da teoria existente, e buscou focar em aspectos ligados ao tema de ferramentas educacionais. O Construcionismo é uma filosofia centrada em apoiar a liberdade do aprendiz, e para isso propõe o uso de ferramentas poderosas que o auxiliem a criar e a pensar sobre essas criações. A programação é uma dessas ferramentas e permite o exercício da criatividade e autonomia por parte do estudante. A programação se dá por diferentes interfaces, e o seu desenvolvimento seguindo princípios fundamentados em teorias como o construcionismo já ocorre desde a linguagem Logo. Esses princípios têm como influência o construtivismo de Piaget, que apontou a necessidade de materiais concretos para apoiar o raciocínio. Neste sentido, os princípios do currículo em espiral de Bruner também influencia a construção de brinquedos de modo que possuam interfaces de programação com níveis crescentes de poder e complexidade. Exemplos são apresentados na Seção 3.3.

2.2 PENSAMENTO COMPUTACIONAL

O Pensamento Computacional é um termo utilizado inicialmente por Papert (1980), mas que tem recebido maior atenção a partir de 2006, quando Jeannette Wing publicou o artigo opinativo intitulado "Computational Thinking" na revista *Communications of the ACM* (WING, 2006). No artigo, Wing define o PC como "[...] conjunto de atitudes e habilidades universalmente aplicáveis que todos, não apenas os cientistas da computação, estariam ansiosos por aprender e usar" (WING, 2006, p.33, tradução nossa). Considerando essa universalidade de aplicações, Wing afirma que o PC deve ser aprendido pelas crianças assim como a escrita, leitura e matemática. Em trabalhos posteriores a autora menciona que o Pensamento Computacional é um processo de pensar um problema para admitir uma solução computacional. Essa solução pode ser gerada por uma máquina, por um humano, ou pela combinação de ambos. Em 2014, Wing acrescenta que o PC não trata apenas de como resolver

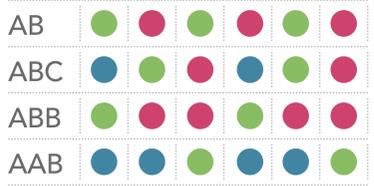
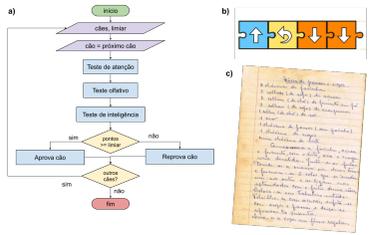
problemas, mas também de como formulá-los (WING, 2014).

Barr e Stephenson (2011) afirmam que uma definição de PC, para ter utilidade, precisa trazer exemplos de como incorporá-lo aos ambientes educacionais. Neste sentido cerca de 700 professores da *International Society for Technology in Education* (ISTE) ISTE (*International Society for Technology in Education* — Sociedade Internacional para Tecnologia na Educação) e da *Computer Science Teachers Association* (CSTA) desenvolveram uma “definição operacional” do Pensamento Computacional. Essa definição caracteriza o PC como um processo de resolução de problemas, que inclui (mas não se limita) a:

- Formular problemas de forma que um computador possa auxiliar a resolvê-los;
- Organizar e analisar dados logicamente;
- Representar dados através de abstrações, como modelos e simulações;
- Automatizar soluções através de pensamento algorítmico (série de passos ordenados);
- Identificar, analisar e implementar possíveis soluções para alcançar a combinação mais eficiente e eficaz de processos e recursos;
- Generalizar e transferir o processo de solução do problema para outros problemas.

Brackmann (2017), a partir da definição de BBC Learning (2015), descreve o PC em quatro pilares (Quadro 1): decomposição, reconhecimento de padrões, abstração e algoritmos. Esses quatro pilares podem ser aplicados para solucionar problemas complexos. O problema pode ser inicialmente quebrado em partes menores e mais simples de compreender e organizar (decomposição). Essas partes podem ser analisadas para verificar se alguma solução existente se aplica ao subproblema atual (reconhecimento de padrões), e também verificar quais informações são relevantes para solucioná-lo (abstração). Por último, um conjunto de passos descreve a solução de cada subproblema (algoritmo). O Quadro 1 exemplifica a aplicação de cada pilar.

Quadro 1. Pilares do Pensamento Computacional.

Pilar	Exemplo	
<p>Decomposição: Separar partes que constituem um todo, potencialmente facilitando a resolução de problemas.</p>	<p>Na escrita de um texto, em vez de escrever o texto completo de uma única vez, a tarefa pode ser decomposta em introdução, desenvolvimento e conclusão. O desenvolvimento pode então ser decomposto em um conjunto de ideias a serem transmitidas com parágrafos. Até mesmo os parágrafos podem ser decompostos, com uma frase inicial, uma ou mais frases de suporte e uma frase de conclusão (DAVIDSON; MURPHY, 2018).</p>	
<p>Reconhecimento de Padrões: Identificar similaridades entre as partes de um problema para resolvê-las com eficiência (BBC Learning, 2015). Na Ciência da Computação os padrões reduzem a complexidade por meio da generalização de soluções para aplicá-las em múltiplas situações (K-12 Computer Science Framework, 2016).</p>	<p>Um exemplo é reconhecer padrões em sequências de cores. A primeira vista são apenas círculos coloridos, mas é possível identificar o padrão sequencial de cores em cada linha. Outros exemplos são identificar rotas comuns entre casa e trabalho para organizar caronas, identificar o padrão de compras de um consumidor, e também reconhecer sintomas similares de doenças entre pacientes.</p>	
<p>Abstração: Destacar detalhes importantes e ignorar detalhes desnecessários (WING, 2008). A abstração elimina os detalhes específicos, inúteis para resolver um problema, e cria uma ideia base da solução denominada modelo (BBC Learning, 2015).</p>	<p>Durante o treinamento de cães farejadores para resgate de seres humanos, um conjunto de testes avalia os animais em quesitos como inteligência, concentração, agilidade e capacidade olfativa. Dados como raça, tamanho, cor ou peso não influenciam na capacidade do cão desempenhar a tarefa em questão. É preciso, portanto, detectar as características relevantes ao problema no universo de características do animal (DAVIDSON; MURPHY, 2018).</p>	
<p>Algoritmos: instruções que demonstram o passo a passo para resolver um problema. Esse conjunto de passos precisa ter um início, um fim, e uma sequência de instruções sem ambiguidade (BBC Learning, 2015).</p>	<p>A representação desse conjunto de passos pode ocorrer por diferentes modos. Uma receita de bolo é um algoritmo, pois define o conjunto de passos para resolver o problema de criar um bolo. Neste caso a representação é textual, porém pode-se representar um algoritmo por fluxogramas, blocos, sequência de símbolos, etc.</p>	

As definições de PC apresentadas estão relacionadas com resolução de problemas gerais, e não se restringem ao contexto da programação de algoritmos. Nessas definições, o conceito de depuração parece ter relevância secundária. Uma possível causa é o fato da depuração estar “embutida” na programação algoritmos, e não ser percebida como uma habilidade útil no dia a dia. McCauley et al. (2008) comenta que até mesmo no campo da programação a depuração é negligenciada, afirmando que o livro de Barnes e Kölling (2002) é um dos poucos a dedicar um capítulo completo sobre depuração. A próxima seção busca esclarecer este conceito.

2.2.1 Depuração

A depuração é um conceito que foi particularmente discutido por pesquisadores e educadores nas décadas de 1970 (MCCAULEY et al., 2008) e 1980 (SIPITAKIAT; NUSEN, 2012). Mesmo já sendo praticada por programadores profissionais, foi o surgimento da linguagem Logo neste período que desencadeou estudos envolvendo depuração por crianças. A recente popularização do PC (ILIC; HASESKI; TUGTEKIN, 2018) e o aumento do contato de crianças com algoritmos tem levado pesquisadores a discutirem o papel da depuração no aprendizado (REPISO; GONZÁLEZ, 2019).

A atividade de depurar está ligada à necessidade de corrigir partes de algoritmos que impedem a execução correta de um programa. No contexto educacional, este desejo cria uma situação propícia ao exercício de um conjunto maior de habilidades, como trabalho em equipe, comunicação e persistência (SIPITAKIAT; NUSEN, 2012). Portanto, pode ser vista como uma atividade de resolver problemas, que envolve observar, comunicar e refletir. A depuração, bem como a ação de ler e acompanhar a execução passo a passo de programas existentes, são consideradas essenciais para aprender programação (MCCAULEY et al., 2008) e contribuir para o desenvolvimento do PC.

Entretanto, Liu et al. (2017) observam que a depuração é um componente do PC negligenciado principalmente nos níveis iniciais de ensino. Os avanços tecnológicos e o entusiasmo com a temática do PC tem levado estudantes iniciarem o aprendizado de programação mais jovens. Ainda assim, afirma que poucos ambientes de programação são projetados com foco em funcionalidades de depuração no nível da Educação Básica, e a maior parte das pesquisas sobre o tema ocorrem com estudantes em nível universitário.

Brennan e Resnick (2012) abordam a depuração em um framework de PC utilizado no contexto do ScratchJr. O framework tem três dimensões: conceitos, práticas e perspectivas. Os conceitos remetem aos conhecimentos utilizados e aprendidos durante a programação, como paralelismo, laços

de repetição e dados. As práticas envolvem a aplicação desses conceitos utilizando de desenvolvimento iterativo, mesclagem de códigos e depuração. Por fim, as perspectivas são formas de se utilizar a tecnologia: como meio de expressão, conexão, e questionamento/mudança da realidade. Usando esse framework, Lye e Koh (2014) revisaram a literatura em busca de trabalhos sobre PC. Eles observam que 85% dos artigos analisaram os aprendizados de PC relacionados a conceitos, e apenas 22% analisaram as práticas. Argumentam que por isso seria necessário mais trabalhos ligados a práticas computacionais como a depuração.

Wong e Jiang (2018) relacionam pensamento algorítmico e depuração como duas sub-habilidades do PC. O pensamento algorítmico seria a capacidade de formular sequências de passos para solucionar um problema independente de computador. A depuração seria aplicada depois do pensamento algorítmico, ao traduzir o algoritmo para uma linguagem de programação e solucionar problemas no código do programa se este não gerar o resultado desejado. Papert, por outro lado, afirma que a depuração independe do computador, pois “estratégias de depuração foram desenvolvidas por estudantes de sucesso muito antes da existência dos computadores” (PAPERT, 1980, p.23). Seria, então, mais um processo mental do que o uso de ferramentas computacionais.

Carver e Klahr (1986) decompõem o processo da depuração em quatro etapas: (i) avaliar o programa, (ii) identificar o *bug*, (iii) localizar o *bug*, (iv) e corrigir o *bug*. Em uma atividade com BPs, por exemplo (ver Figura 5), a criança executa o algoritmo e verifica se alcançou o resultado esperado. Caso contrário, precisa identificar o erro, por exemplo, se o brinquedo está no local errado, se virou para o lado errado, etc. Ainda na identificação, constrói hipóteses causas do problema. Faltou adicionar um comando de giro? O comando de giro está errado? A terceira etapa passa a ter contato com o código, onde a criança identifica qual peça pode ser o *bug*. Por fim, na última etapa a criança modifica o programa, adicionando ou removendo a peça apontada na etapa 3 e o ciclo recomeça com uma nova execução.

Seguindo esse modelo, Carver e Klahr (1986) avaliaram o aprendizado de depuração por crianças entre 7 e 8 anos. As crianças trabalharam em pares durante 22 horas de curso sobre a linguagem Logo. Os pesquisadores aplicaram dois testes em dupla e um individual para avaliar a habilidade de depuração, interpretação de código, escrita de código e uso do ambiente Logo. Os resultados indicaram que as crianças depuraram somente quando lhes era solicitado, e que a estratégia utilizada para encontrar erros foi procurar linha por linha, sem uma estratégia eficiente. As crianças também preferiram apagar o programa em vez de depurar. Por fim, concluem que depurar é uma atividade complexa, que exige memória de trabalho para observar o programa criado e os problemas

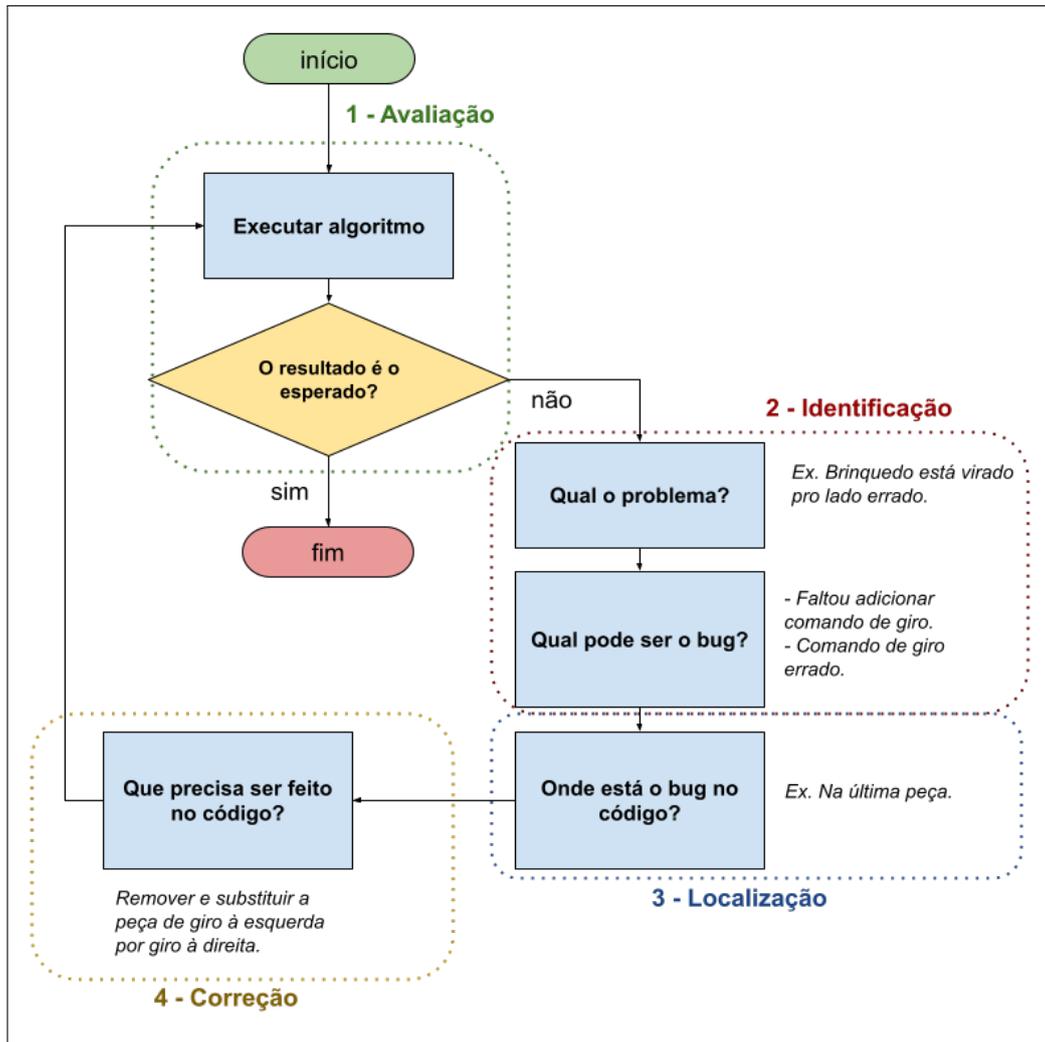


Figura 5. Processo de depuração proposto por Carver e Klahr (1986) no contexto de um BP.
Fonte: O autor.

existentes. Além disso, menciona que o fato do erro ser punido pelo sistema educacional leva a criança a tentar eliminá-lo (apagar todo o programa) em vez de resolvê-lo.

2.2.2 Depuração e visibilidade

Em uma revisão sistemática, McCauley et al. (2008) apresenta modelos de depuração definidos Vessey (1985) e Katz e Anderson (1987). Vessey (1985) define depuração em cinco passos: (i) determinar o problema comparando o resultado errado e o correto; (ii) *obter familiaridade com a estrutura do programa*; (iii) explorar a execução do programa; (iv) levantar hipóteses para a causa do erro; e (v) reparar o erro. Katz e Anderson (1987) definem quatro passos: (i) entender o sistema, (ii) testar o sistema, (iii) *localizar o erro*, e (iv) reparar o erro.

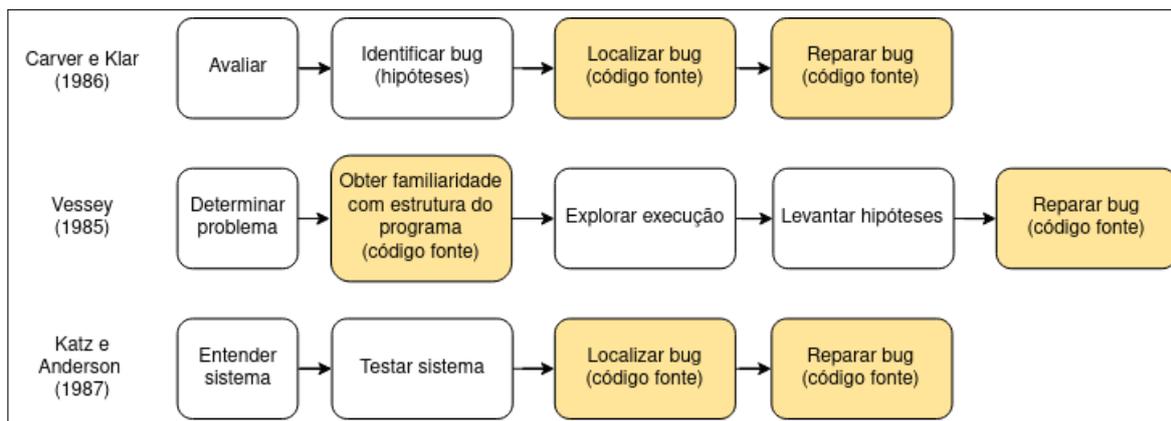


Figura 6. Modelos de depuração.

Fonte: O autor.

Estes modelos têm em comum um passo de observar a estrutura do algoritmo para localizar o erro (destacados na Figura 6). A localização do erro presume um código-fonte a ser analisado. Essa necessidade está ligada ao que Norman (1990) apresenta como um dos princípios mais importantes do design: a visibilidade. Esse princípio permite o “mapeamento entre as ações pretendidas e as ações concretas” (NORMAN, 1990). Para isso, as ações possíveis de se realizar com uma interface devem estar visíveis e seus efeitos devem ser óbvios e imediatos.

Interfaces de programação que não atendem o princípio da visibilidade dificultam a depuração. Este problema aparece em ferramentas projetadas para crianças, como a Bee-Bot. A sua interface tem botões direcionais e um botão de limpeza de memória, que apaga os comandos inseridos. O brinquedo não tem um *display* para mostrar os comandos digitados. Ao não ver os comandos, as crianças esquecem de limpar a memória e então pensam estar construindo um programa novo (RAABE et al., 2015). A falta de visibilidade impede perceber que os comandos inseridos anteriormente ainda estão presentes e os novos comandos são adicionados no final da sequência.

Durante este processo ocorre um fenômeno mencionado por Norman (1990) como a formação do *modelo mental* do usuário. O modelo mental é a compreensão que o usuário tem sobre quais ações deve comunicar a um sistema para obter um resultado desejado. A construção do modelo mental depende que o usuário perceba o estado atual do dispositivo, e como ele se modifica conforme as entradas informadas. Quando a interface com o dispositivo não é clara ou a apresentação do estado ambíguo, o comportamento do dispositivo pode não se adequar ao modelo mental construído pelo usuário.

A questão é que, quando o inesperado acontece, o cérebro humano tenta explicar e acomodar

essa nova situação. No caso da Bee-Bot, ao vê-la executar movimentos diferentes dos últimos comandos programados — que ocorre pela diferença entre o modelo pensado pela criança e o estado real do brinquedo — as crianças tendem a explicar que “abelha” está “tomando suas próprias decisões”. A criança, portanto, adapta o seu modelo mental para explicar o comportamento do dispositivo computacional quando este não executa a sequência de passos que recém-programados.

Uma alternativa para interfaces com o problema de falta de visibilidade são os ambientes de programação em blocos (Subseção 2.3.3). Wong e Jiang (2018) atribuem a melhoria na habilidade de depuração de crianças do quinto ano do Ensino Fundamental às “características de visualização da programação em blocos, pois os efeitos do programa podem ser vistos diretamente”. Bers (2019) fala sobre as vantagens de visualização das interfaces de programação em blocos, neste caso blocos tangíveis:

As crianças podem ver o que não funciona e podem ajudar a consertar. [...] A visibilidade adicional do código pode chamar a atenção para propriedades e conceitos anteriormente negligenciados, como elegância, a importância da depuração e teste para casos extremos e outras situações incomuns. Em outras palavras, o código tem o potencial de se tornar um objeto de conversa e atenção de uma forma que não poderia ser antes. (BERS, 2019, p.14, tradução nossa)

Além de promover a visibilidade, a programação em blocos evita problemas de interação presentes em outras categorias de interfaces, como a programação textual. Esta, apesar de também mostrar os comandos, não é acessível para crianças devido a erros de sintaxe e à necessidade de lembrar o que é preciso digitar. Os blocos, por outro lado, estão sempre disponíveis, geralmente agrupados por similaridade, de modo que não seja necessário lembrar sequências de caracteres para digitar num teclado. Além disso, o formato dos encaixes sugere como relacionar os blocos de modo coerente.

A visibilidade da programação em blocos também favorece a interação social. Os blocos visíveis podem ser apontados, manipulados, e tornam o código um objeto de conversa. O mesmo ocorre com a depuração, que pode ser uma atividade estabelecida socialmente com falas, gestos e olhares. Neste sentido, os blocos são artefatos com papel central (HEIKKILÄ; MANNILA, 2018) ao depurar. Esta ligação entre blocos e depuração remete ao princípio da linguagem Logo, de permitir depurar o que se passa na mente da criança (SOLOMON et al., 2020). Os artefatos compartilhados socialmente se tornam mais um meio de comunicação para o adulto compreender o algoritmo que a criança deseja expressar. Deste modo pode auxiliá-la a desenvolver seu modelo mental de como construir um algoritmo e resolver um problema.

2.3 INTERFACES DE PROGRAMAÇÃO PARA CRIANÇAS

Aprender programar algoritmos parece ser uma tarefa difícil, mesmo para adultos. Analisando cursos de graduação em Computação de universidades dos Estados Unidos, Israel, Polônia e Austrália, McCracken et al. (2001) concluíram que os estudantes não possuíam as habilidades de programação esperada pelos professores, indicando dificuldade no aprendizado de algoritmos. Hoed (2016) cita a reprovação na disciplina de Algoritmos entre as causas de evasão em cursos de Computação.

Mesmo sendo um conteúdo complexo, crianças podem aprender noções básicas algoritmos. Sheehan et al. (2019), observaram 31 crianças de 4,5 a 5 anos, durante a interação com um aplicativo de programação, e concluem que elas conseguiram produzir e compreender algoritmos básicos para definir o comportamento de um personagem. Em outro estudo, com 53 crianças de 4 a 6 anos, Bers et al. (2014) identificam que 75% das mesmas selecionaram e sequenciaram corretamente instruções ao programar um veículo robótico.

Os benefícios de aprender algoritmos transcendem os campos estritamente tecnológicos. Conforme lembram Çiftci e Bildiren (2020), os algoritmos possuem conceitos que se intersectam com os da matemática, como variáveis, laços de repetição e condicionais. Esses autores também afirmam que programar pode auxiliar estudantes a analisarem seu modo de raciocínio, reorganizar tarefas seguindo um processo de resolução de problemas e desenvolver habilidades não-verbais. Em experimento, os autores verificaram um aumento significativo dessas habilidades em crianças de 4 a 5 anos que participaram de um curso de programação durante 8 semanas.

Para Papert (1980),

Quando a criança aprende a programar, o processo de aprendizado é transformado. Ele se torna mais ativo e autodirigido. Em particular, o conhecimento é adquirido para um propósito pessoal reconhecido. A criança faz algo com ele. O novo conhecimento é uma fonte de poder sendo experimentado como tal desde o momento em que começa a tomar forma na mente da criança (PAPERT, 1996, p.21, tradução nossa).

Uma das primeiras experiências envolvendo crianças e programação é atribuída a Papert e Cynthia Solomon em 1968. Ambos foram até uma escola de um subúrbio de Boston ensinar programação para jovens. As interações ocorreram através de terminais, que recebiam comandos em Logo e executavam em um computador do laboratório a alguns quilômetros de distância.

Conforme novas versões da linguagem eram desenvolvidas, pesquisadores as testavam em

escolas ao ensinar e observar o uso da linguagem em sala de aula (BERS, 2018). Solomon destaca o papel da participação das crianças durante o desenvolvimento da linguagem, dando *feedback* e influenciando diversas questões de design:

O trabalho iniciou o que se tornou uma linguagem poderosa, flexível e usável para crianças e destacou a facilidade com que os aprendizes adquiriram experiência sobre a ferramenta. O trabalho é também um importante exemplo de envolvimento das crianças no design de uma nova tecnologia e de como a tecnologia pode se beneficiar da expertise e do *feedback* das crianças. Cada vez que Seymour e eu trabalhamos com as crianças, LOGO foi radicalmente redesenhada incorporando o seu *feedback*.

A linguagem LOGO inicialmente permitia manipular palavras, e as crianças construíam poemas e geradores de frases. Papert, um matemático, percebeu a necessidade transcender o campo das palavras e de permitir às crianças brincarem com formas, ângulos e desenhos. Dessa percepção nasceu a ideia de uma tartaruga capaz de se mover e desenhar figuras geométricas. Essa tartaruga foi introduzida primeiro virtualmente, mas depois tomou formas físicas e originando os primeiros brinquedos programáveis.

2.3.1 Brinquedos Programáveis

Brinquedos programáveis podem ser caracterizados, em sua maioria, como robôs sobre rodas que recebem comandos e executam em forma de movimentos¹. O que os diferencia das outras ferramentas de programação é a aparência lúdica, adaptada ao imaginário do público infantil (RAABE et al., 2017). Pesquisas mencionam que os brinquedos programáveis podem promover habilidades relacionadas do PC (REPISO; GONZÁLEZ, 2019; BERS, 2018; PUGNALI; SULLIVAN; BERS, 2017; BERS et al., 2014), como pensamento algorítmico, reconhecimento de padrões, abstração e decomposição.

A interface intuitiva é o que permite que crianças possam programar esses brinquedos. Diferente de teclado de computador, eles geralmente têm poucos botões e não exigem alfabetização. O desenvolvimento dessas interfaces se inicia com o ambiente Logo, e diversas alternativas de interação têm surgido nos últimos anos (CATLIN et al., 2018).

Hamilton et al. (2020) apresentam 30 brinquedos e os classificam segundo suas características físicas em seis categorias: (i) jogos de tabuleiro ou livros; (ii) eletrônicos não robóticos; (iii) robôs controlados por tela; (iv) robôs operados por botões; (v) robôs com interface tangível; e (vi) híbridos

Figura 7.

¹ Há brinquedos que não são robóticos, como o Robot Turtles, um jogo de tabuleiro com conceitos de programação.

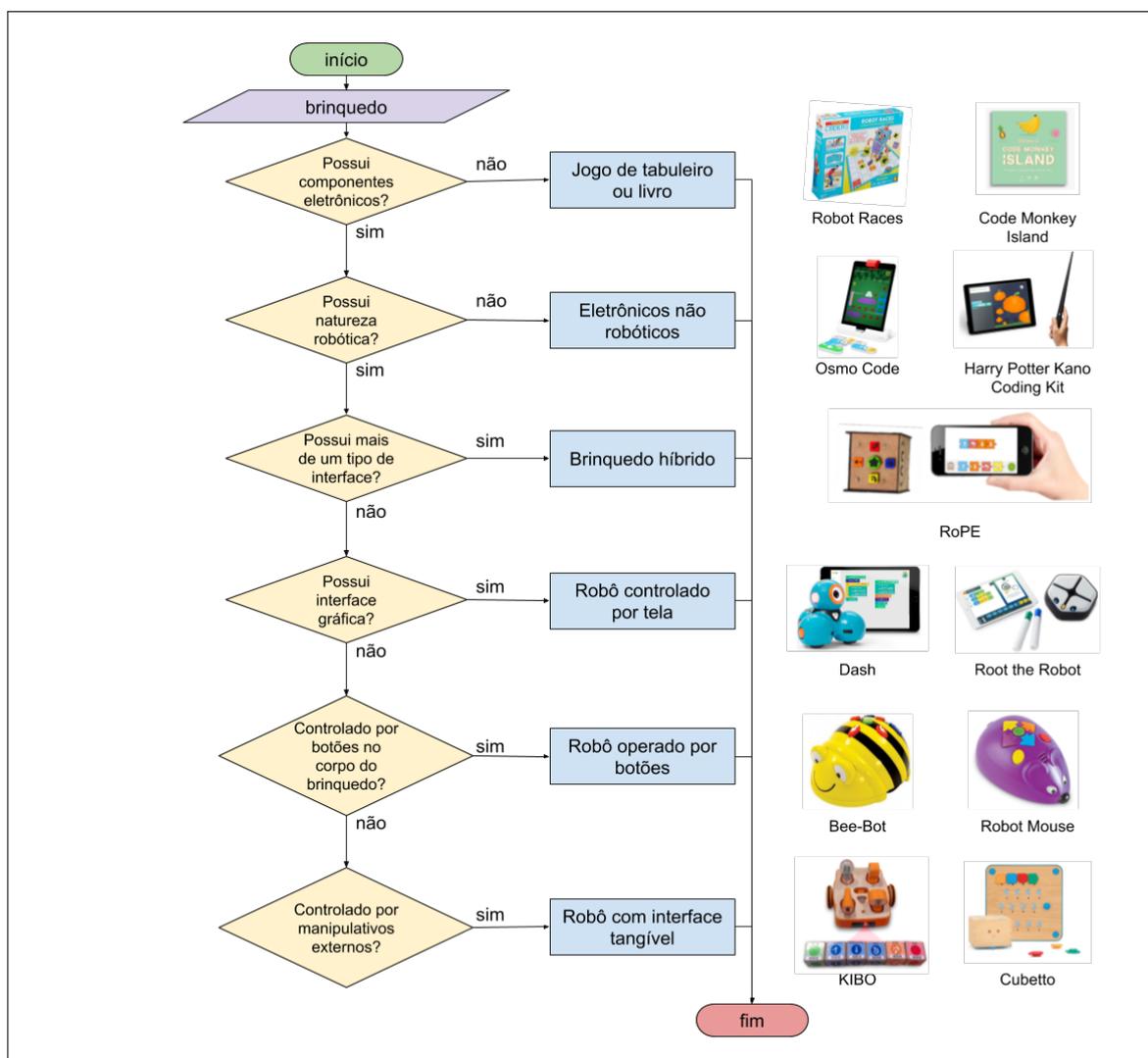


Figura 7. Classificação de brinquedos programáveis por características físicas.

Fonte: Adaptado de Hamilton et al. (2020).

Em outra revisão, Yu e Roque (2019) também analisam um conjunto de 30 brinquedos programáveis e kits de robótica focados em crianças de até 7 anos. A análise ocorre sob as perspectivas de design, Pensamento Computacional, expressividade e suporte a domínios².

Quanto ao design, Yu e Roque (2019) distinguem três categorias de kits: físicos, virtuais e híbridos. As três categorias têm o comum a utilização de blocos de código relacionados a movimentos, com cada bloco tendo uma cor diferente para auxiliar na distinção dos diferentes comandos. Esses blocos comumente tem o formato de peças de quebra-cabeça. Os kits físicos, como KIBO e Cubetto, possuem todos os componentes tangíveis, geralmente incluindo um robô com rodas, um conjunto de

² A expressividade corresponde à variedade de atividades possibilitadas pelo kit, e o suporte a domínios se refere ao conjunto de assuntos/conteúdos passíveis de serem apresentados no contexto dessas atividades.

blocos de código e materiais de suporte como mapas, livros e materiais de personalização. Alguns robôs são controlados por botões, mas a maioria tem blocos de códigos separados.

Os kits virtuais são aqueles que não possuem partes tangíveis. Exemplos são aplicativos de *smartfone* e jogos de computador, que possibilitam construir cenas, programar personagens e resolver problemas ao arrastar e soltar blocos de código virtuais. O Scratch e o ScratchJr são aplicativos que permitem construir cenas e programar personagens. Essa categoria remete às primeiras experiências de Papert durante o desenvolvimento da linguagem LOGO, que permitia definir o comportamento da tartaruga virtual. A diferença é que as ferramentas atuais priorizam o uso de linguagens de blocos.

A terceira categoria são os kits híbridos, que unem partes tangíveis e virtuais. Yu e Roque (2019) destacam duas classes de kits híbridos: aqueles com blocos físicos que controlam personagens virtuais (brinquedos da empresa Osmo, por exemplo³), e aqueles que possuem blocos virtuais para controlar personagens físicos.

Quanto ao Pensamento Computacional, Yu e Roque (2019) relacionam atividades possibilitadas pelos BPs e kits capazes de promover os conceitos e práticas ligadas ao PC definidas por Brennan e Resnick (2012). Os autores identificam que as atividades com BPs podem abordar todos os conceitos e práticas do PC. Há poucos kits, porém, que suportam as práticas de reutilização/cominação e abstração/modularização.

Quanto à expressividade, ou seja, as atividades que os brinquedos possibilitam, os autores citam três principais modalidades. A primeira corresponde a mover o brinquedo/kit por um caminho ou sobre um mapa para coletar objetos. A segunda modalidade é a contação de histórias: os brinquedos se movem sobre cenários, e as crianças podem criar histórias para os mesmos em conjunto com a família. Há também uma terceira atividade: decorar e personalizar a aparência do brinquedo.

Quanto ao suporte a domínios, Yu e Roque (2019) mencionam o desenvolvimento de narrativas, conceitos matemáticos e conceitos de engenharia. O desenvolvimento de narrativas está ligado às atividades de contação de histórias. O Scratch, por exemplo, possibilita criar narrativas por meio da programação de personagens e definição de cenários. Conceitos matemáticos aparecem ao definir distâncias de movimentos, e operações aritméticas simples também estão presentes em alguns brinquedos. Os conceitos de engenharia são abordados em brinquedos onde é possível unir partes diversas e observar efeitos, como o caso do Cubelets⁴, que permite à criança unir cubos de luminosidade, sensores

³ <<https://www.playosmo.com/en/coding/>>

⁴ <<https://www.modrobotics.com/#whatr-cubelets>>

Quadro 2. Como brinquedos programáveis e kits promovem conceitos e práticas computacionais.

Conceitos	
Sequenciamento	Criar uma sequência de código para programar o movimento ou outros efeitos em robôs físicos/virtuais.
Repetições	Encapsular uma sequência de comandos em blocos de repetição para executar repetidas vezes.
Eventos	Executar ações ao apertar botões específicos, ou programar efeitos durante interações entre brinquedos (por exemplo, Dash e Dot executam ações quando se aproximam um do outro).
Paralelismo	Controlar o movimento de diversos robôs/sprites simultaneamente, ou programar efeitos simultâneos de movimento, luzes e sons.
Condicionais	Programar ações dependentes de eventos disparados por sensores ou mapas.
Operadores	Possibilitar realizar operações matemáticas simples.
Dados	Ajustar parâmetros como distância do movimento, rotação, etc.
Práticas	
Desenvolvimento iterativo e incremental	Permitir erros e tentativas ilimitados. Constantemente revisar e adicionar novos blocos de código (ex. ScratchJr).
Teste e depuração	Testar continuamente o código durante o desenvolvimento e depurar se algo não funciona.
Reutilização e combinação	Construir projetos partindo de projetos anteriores (ex. Scratch). A maioria dos kits não suporta essa prática.
Abstração e modularização	Construir blocos de código que podem ser chamados de outros locais (ex. Cubetto). Poucos kits suportam essa prática.

Fonte: Adaptado de Yu e Roque (2019).

e movimentos.

As classificações de Hamilton et al. (2020) e de Yu e Roque (2019) se diferem quanto aos critérios utilizados. Enquanto Hamilton et al. (2020) não inclui em sua lista os kits robóticos e softwares exclusivamente gráficos, Yu e Roque (2019) inclui ambos. Ambos os trabalhos, porém, focam no público infantil, e as análises das interfaces e dos conceitos promovidos são úteis para definir as características do brinquedo RoPE.

2.3.1.1 RoPE - Robô Programável Educacional

O RoPE é um brinquedo programável desenvolvido para crianças a partir de 3 anos, que busca estimular o Pensamento Computacional. Ele permite à criança inserir comandos que ele então executa como movimentos sobre um tapete temático. A construção tem preocupações de design relacionadas ao público e ao ambiente. O público tem coordenação motora em desenvolvimento e isso reflete o tamanho dos botões. As cores dos botões facilitam a comunicação das professoras para indicarem um botão específico durante alguma explicação no ambiente educacional.

O RoPE pode ser operado de duas formas: botões na parte superior e um aplicativo de *smart-*

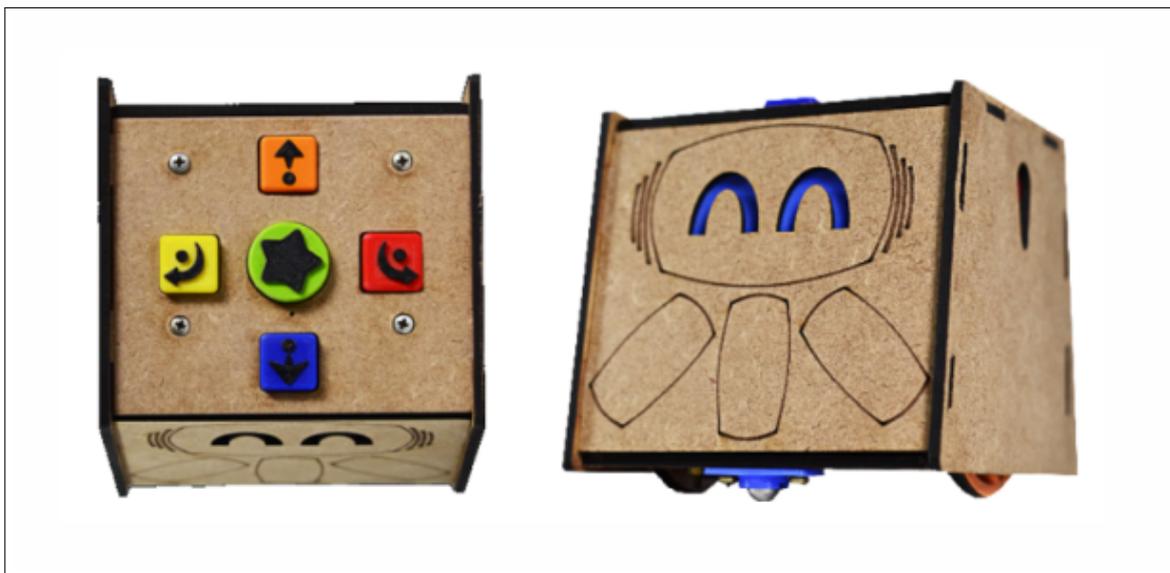


Figura 8. Visão superior e frontal do RoPE.

Fonte: Smartfun Brasil (<<https://www.smartfun.com.br>>).

phone. A primeira interface é formada por 5 botões que fazem parte do brinquedo. São eles: (1) amarelo para girar 90 graus à esquerda, (2) vermelho para girar 90 graus à direita, (3) azul para andar à frente (4) laranja andar para trás, e (5) verde para executar instruções inseridas. A execução de cada instrução provoca, além do movimento, a emissão de um som característico e uma luz da mesma cor do botão gerador da instrução, para haver uma tríade cor-movimento-som. Um som de finalização indica quando o brinquedo executou todas as instruções (RAABE et al., 2017).

A segunda interface é um aplicativo de celular que permite programar o RoPE a distância (Figura 9). O programa aparece na tela em formato de peças de quebra-cabeça conectadas. A comunicação Bluetooth entre o brinquedo e o aplicativo possibilita manter as peças e as instruções da memória em sincronia. A reordenação das peças no aplicativo é replicada no brinquedo e o pressionamento de um botão do brinquedo é replicado nas peças na tela do celular. Considerando a possibilidade de programar por duas interfaces, as classificações de Hamilton et al. (2020) e Yu e Roque (2019) posicionam o RoPE como um brinquedo híbrido.

O Quadro 3 classifica o RoPE quanto aos aspectos do Pensamento Computacional. Os conceitos abordados pelo brinquedo são sequenciamento, eventos e dados. O brinquedo aborda diretamente apenas sequenciamento e eventos, porém as professoras podem abordar o conceito de dados durante o uso do brinquedo, ao definir distâncias e posições a serem alcançadas. O brinquedo não abrange todas as práticas e conceitos, pois o público-alvo exige um design simples de usar.

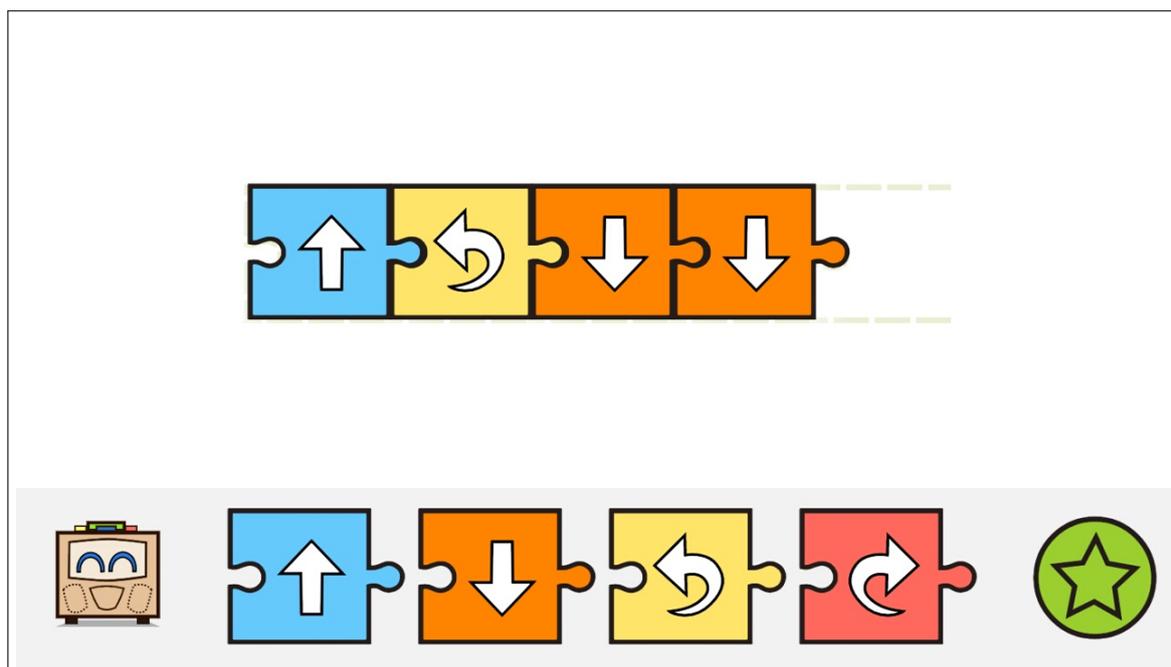


Figura 9. Aplicativo para programar o RoPE.

Fonte: O autor.

Quadro 3. Conceitos e práticas computacionais e sua promoção pelo brinquedo RoPE.

Conceitos	
Sequenciamento	Criar uma sequência de comandos para guiar o movimento do RoPE sobre um mapa.
Repetições	Conceito não abordado diretamente
Eventos	O evento de clicar na estrela inicia a execução dos comandos inseridos previamente.
Paralelismo	Conceito não abordado diretamente
Condicionais	Conceito não abordado diretamente
Operadores	Conceito não abordado diretamente
Dados	Estimar o número de passos para alcançar uma posição no mapa.
Práticas	
Desenvolvimento iterativo e incremental	Não abordado diretamente, pois as instruções são apagadas do brinquedo ao final de cada execução.
Teste e depuração	Testar continuamente o código durante o desenvolvimento e depurar se algo não funciona.
Reutilização e combinação	Conceito não abordado diretamente
Abstração e modularização	Conceito não abordado diretamente

Fonte: O autor.

Além de aspectos ligados ao PC, o RoPE suporta domínios multidisciplinares através de tapetes temáticos, nos quais as crianças têm contato com letras do alfabeto (Figura 10a), temáticas dos ambientes rurais (Figura 10b) e urbanos (Figura 10c). Pinheiro (2016) desenvolveu um software que permite à criança gerar seus próprios mapas e percebeu mais engajamento das crianças ao usarem seus próprios mapas. Esses tapetes também podem ser criados manualmente, como o tapete criado pelas crianças da Figura 10a. Além de criar os próprios tapetes Martins (2016) também menciona a possibilidade de customizar a carcaça do brinquedo. Então, apesar de a interface do brinquedo favorecer raciocínio matemático de estimar quantidades e movimentos, a customização e os tapetes possibilitam criar micromundos infinitos associados a qualquer temática.



(a) Letras do alfabeto



(b) Ambiente rural



(c) Ambiente urbano

Figura 10. Tapetes temáticos.
Fonte: (RAABE et al., 2017)

2.3.2 Interfaces Tangíveis

Como abordado na Subseção 2.3.1, as interfaces tangíveis são interfaces utilizadas pelos brinquedos de programar. O conceito de interfaces tangíveis, entretanto, é amplo e tem fronteiras difusas (FALCÃO, 2007). A ideia inicial sobre tangibilidade é que apresenta formas físicas de interação, que permitem o contato da mão com objetos concretos. Essa interação é considerada mais intuitiva e capaz de estimular colaboração e diversão (ZUCKERMAN; GAL-OZ, 2013). A tangibilidade dos blocos de encaixar, como LEGO, por exemplo, leva crianças a permanecerem horas manipulando-os e testando possibilidades de construções.

A ideia de manipulação de objetos concretos não delimita completamente o que são interfaces tangíveis. Dado que são manipulados diretamente, “teclados são TUIs?” (FALCÃO, 2007, p.30). Neste sentido, diferentes taxonomias buscam organizar a definição de interfaces tangíveis (FISHKIN, 2004;

HORN; BERS, 2019). Fishkin (2004), por exemplo, observa a sequência de ações do usuário durante uma interação. Em toda interação, o usuário manipula uma interface e em seguida observa o resultado da manipulação. A taxonomia considera a distância entre os pontos de manipulação e o ponto de expressão do resultado, ao que denomina “incorporação”. Quanto mais próximas estão a manipulação e o resultado, maior a incorporação. Há quatro níveis de incorporação das interfaces tangíveis (FISHKIN, 2004):

Completa: O ponto de manipulação é o mesmo que expressa o resultado. Um exemplo é o ábaco.

Próxima: A expressão do resultado ocorre próxima do ponto de manipulação. Uma caneta é um exemplo.

Ambiental: A expressão do resultado está “ao redor” do usuário. Um exemplo seria uma sala com projeção nas paredes.

Distante: O ponto de manipulação e a expressão do resultado ficam em salas separadas, ou a metros de distância. O controle remoto é um exemplo.

Horn e Bers (2019) apresentam a ideia de linguagens tangíveis de programação. Neste caso, elementos tangíveis são agrupados para construir um significado, que no caso correspondem a um algoritmo. Há três categorias de linguagem tangíveis:

Linguagens de blocos inteligentes: Blocos físicos com componentes eletrônicos internos. Eles conseguem armazenar instruções digitais e executá-las. Um exemplo é o Cubelets.

Linguagens de demonstração: O usuário demonstra ao dispositivo o que deve ser executado e então ele reproduz. A demonstração pode ocorrer com gestos, sons ou entradas digitais (botões).

Linguagens externamente compiladas: A representação do algoritmo ocorre por símbolos sem componentes eletrônicos. Algum agente externo, como uma câmera, capta os símbolos e os transmite como entradas digitais que podem ser convertidas em um programa. Exemplos são blocos de madeira, como os utilizados pelo brinquedo KIBO: um escaner lê cada bloco e o transmite ao brinquedo. Ao escanear uma sequência válida o brinquedo pode executar o algoritmo.

Por fim, Falcão (2007) responde a própria pergunta: teclados e mouse não são TUIs, pois não carregam um significado. Ao contrário, são dispositivos genéricos capazes de gerar significados

variados. Por outro lado, o ábaco é uma interface tangível, pois traz incorporado o significado numérico. Blocos de madeira, como os usados pelo KIBO, é uma linguagem externamente compilada, que pode ser organizado para formar um algoritmo.

2.3.3 Programação em Blocos

Programação em blocos tem se tornado um meio comum de iniciar o aprendizado de programação (WEINTROP, 2019). Ferramentas como o Scratch, ScratchJr e o Code.org utilizam essa abordagem. No Brasil, mais de 40% das iniciativas para ensino de programação adotou ambientes baseados em blocos para introduzir os primeiros conceitos (BORDINI et al., 2016).

A programação em blocos se caracteriza por blocos a peças de quebra-cabeça, que encaixados representam um algoritmo. As cores e o formato indicam quando e como cada bloco pode ser usado. Os encaixes entre os blocos só permitem unir blocos que fazem sentido estarem unidos, evitando erros de sintaxe (WEINTROP, 2019). Além disso, o usuário pode explorar os blocos disponíveis em vez de lembrar palavras a serem digitados.

Outro aspecto da programação em blocos é a exigência ou não de alfabetização. Há blocos que contém texto, o que aumenta a complexidade para crianças em processo de alfabetização (FLANNERY et al., 2013). Um exemplo é o Scratch (Figura 11a). O ScratchJr, por outro lado, é um ambiente de programação desenvolvido para crianças não alfabetizadas. Enquanto o Scratch usa texto, o ScratchJr tem ícones em seus blocos (Figura 11b).



(a) Blocos do Scratch



(b) Blocos do ScratchJr

Figura 11. Blocos do Scratch e do ScratchJr.

Além dos ambientes gráficos, a programação em blocos também existe em interfaces tangíveis. Os blocos gráficos mitigam a necessidade de digitação, porém ainda dependem do mouse ou telas sensíveis ao toque para arrastar e soltar os blocos. Os blocos físicos eliminam essa necessidade, permitindo a manipulação direta e natural. O Osmo (Figura 12a) é um exemplo de uso de blocos

tangíveis de programação. Os blocos são identificados pela câmera de um tablet, e um personagem na tela executa os comandos programados. O Cubetto, já apresentado, utiliza blocos encaixados em um painel de madeira (Figura 12b).



(a) Osmo

Fonte: Tangible Play <<https://www.playosmo.com>>



(b) Cubetto

Fonte: PrimoToys <<https://www.primotoys.com>>

Figura 12. Blocos tangíveis.

Fonte: O autor.

Uma vantagem do painel do Cubetto é indicar, com luzes, qual bloco está sendo executado em cada momento. Isso permite acompanhar a execução, o que é um processo de depuração. Possíveis desvantagens do painel são a limitação do número de encaixes disponíveis e a necessidade de um hardware específico.

2.3.4 Marcas Fiduciais

Interfaces tangíveis aplicam diferentes técnicas de captura de blocos físicos. O Cubetto depende de um hardware específico para identificar os blocos encaixados em um painel, e o Osmo utiliza visão computacional. Uma alternativa que pode facilitar a demarcação de objetos físicos na construção de interfaces tangíveis, e também na construção de aplicações de Realidade Aumentada (Subseção 2.3.4), são as marcas fiduciais.

Marcas fiduciais (Figura 13) são padrões de figuras desenvolvidas para facilitar a identificação e localização de pontos de referências em imagens. Sistemas de marcas fiduciais são compostos por um modelo de marcas e um algoritmo capaz de identificá-las. Apesar de o campo da visão computacional

ter evoluído a ponto de não depender de marcas artificiais para detectar objetos, o uso de marcas fiduciais pode aumentar a confiabilidade e velocidade de processamento em sistemas (FIALA, 2010).

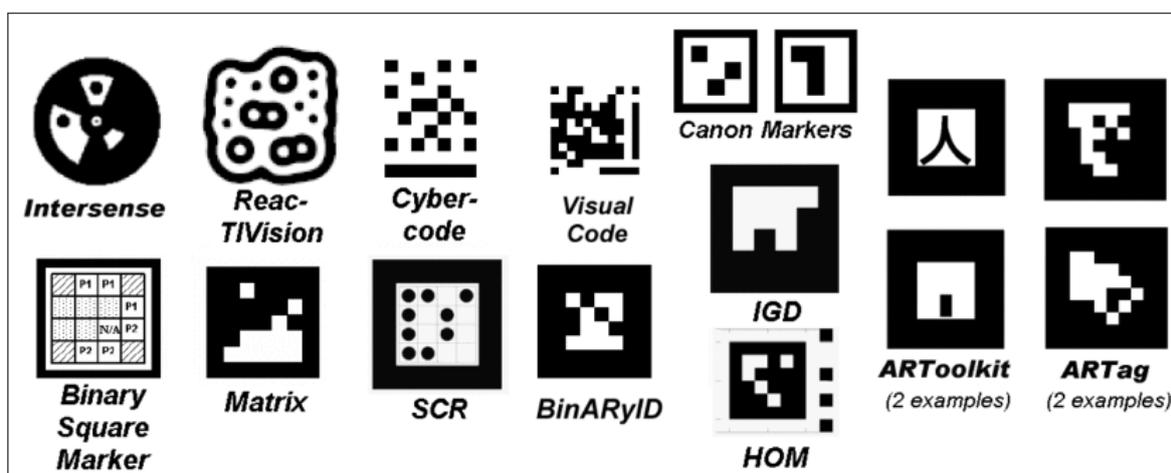


Figura 13. Exemplos de marcas fiduciais.

Fonte: Fiala (2010).

Códigos de barra e QR Codes não são considerados marcas fiduciais. O objetivo destes é carregar informações e não tem dados suficientes para permitir a sua localização. Entretanto, o uso de padrões bitonais (geralmente preto e branco) dos QrCodes e códigos de barra é aproveitado nas marcas fiduciais, dado que isso facilita a detecção quando há variações de luminosidade.

TopCodes⁵ é uma biblioteca de visão computacional, de código aberto, projetada para permitir rápida detecção de marcas fiduciais. Além da identificação e da localização, a biblioteca informa o diâmetro e a orientação de cada marca. Hu et al. (2015) implementaram uma linguagem de blocos tangíveis para programar o personagem do ambiente Osmo (Figura 14a). Viana e Raabe (2018) implementam um ambiente de algoritmos sonoros para deficientes visuais, onde cada marca fiducial representou um instrumento musical (Figura 14b).

Uma desvantagem dessa abordagem é o fato de o usuário encobrir as marcas durante a manipulação. A não detecção de uma peça pode prejudicar o desempenho de sistemas que precisam detectar objetos constantemente. Esse problema, entretanto, não é relatado nos trabalhos mencionados.

2.4 REALIDADE AUMENTADA

A Realidade Aumentada (RA) é uma tecnologia que tem se popularizado nos últimos dez anos. Tem aplicações em inúmeros campos, entre os quais a medicina, a indústria, educação e entretenimento

⁵ <<http://users.eecs.northwestern.edu/~mhorn/topcodes/>>



(a) Tablet captura marcas fiduciais
Fonte: Hu et al. (2015).

(b) Bateria de algoritmos sonoros tangível
Fonte: Viana e Raabe (2018).

Figura 14. Exemplos de uso da biblioteca TopCodes.

(MEKNI; LEMIEUX, 2014). Apesar da popularização recente, as primeiras experiências com RA são atribuídas a Ivan Sutherland, que na década 1960 criou um dispositivo com pequenas telas próximas aos olhos, que permitiam ver o mundo real e adicionavam objetos virtuais adaptáveis às mudanças de perspectiva do usuário. Ainda hoje a RA se caracteriza por complementar a percepção do usuário ao associar elementos virtuais sobre elementos reais (PARVEAU; ADDA, 2018). Além da RA, existem outras técnicas que mostram elementos virtuais e proporcionam experiências realistas de interação, como a Realidade Virtual (RV). Essas técnicas se diferenciam pela quantidade de informação real ou virtual acessada.

Para melhor compreender essa associação Milgram et al. (1994) propõem o conceito de *Virtual Continuum* (Figura 15). Esse conceito tem de um lado a realidade, e do outro a virtualidade. A realidade está restrita a condições físicas, como gravidade e velocidade. A virtualidade, por outro lado, pode ter suas próprias regras. Entre os dois opostos há um intervalo que mistura real e virtual, denominado Realidade Mista (RM). A RA se encontra à esquerda neste intervalo, pois a maior parte da informação acessada pelo usuário existe no ambiente real, e apenas uma camada de objetos virtuais é adicionada. O conceito também apresenta a virtualidade aumentada, que seria a Realidade Virtual. Por ter a maior parte do conteúdo gerado pelo computador e apenas preservar alguns aspectos da realidade, a RV se encontra à direita no intervalo da Realidade Mista.

Azuma et al. (2001) acrescenta outros dois princípios além da combinação de objetos virtuais e reais. Primeiro, o sistema deve responder em tempo real, de modo que as mudanças ocorram



Figura 15. *Continuum* de realidade-virtualidade de Milgram (1994).

Fonte: Adaptado de Milgram et al. (1994).

instantaneamente. Qualquer modificação de perspectiva ou de posicionamento dos objetos reais deve ser imediatamente refletida nos objetos virtuais percebidos pelo usuário. O segundo princípio define que deve haver um alinhamento dos objetos virtuais com os reais, denominado *registro*. Ou seja, o posicionamento dos objetos virtuais deve fazer sentido em relação ao mundo real.

2.4.1 Tecnologias para Criar Realidade Aumentada

Para ser viável, a RA depende de um conjunto de tecnologias para mostrar objetos virtuais e rastrear os objetos reais. A Subseção 2.4.1.1 apresenta diferentes categorias de displays e a seção Subseção 2.4.1.2 cita tecnologias de rastreamento/registro.

2.4.1.1 Displays

Um dos objetivos da RA é produzir integrações de modo que o usuário não consiga distinguir o real do virtual. Diferentes displays têm servido a este propósito. Eles são responsáveis por dispor o objeto virtual em algum ponto entre a retina do observador e o ambiente real. Eles podem estar ou não acoplados ao observador, ser de uso individual ou permitir interações em grupo. Há três classes principais de dispositivos utilizados para mostrar os objetos virtuais: os *head-worn displays*, os *handheld displays* e *projection displays* (AZUMA et al., 2001).

Head-worn displays

Os *head-worn displays* (visores de cabeça) são usados na cabeça do usuário e mostram imagens em visores próximos aos olhos. Essa abordagem foi utilizada por Ivan Sutherland na primeira aplicação de RA. Esses visores são comumente de dois tipos. O primeiro tipo é um visor transparente, como uma lente de óculos, que permite ver o mundo real e também mostra os objetos virtuais. O segundo tipo são visores como telas de LCD. Nessa abordagem uma câmera capta o mundo real e transmite

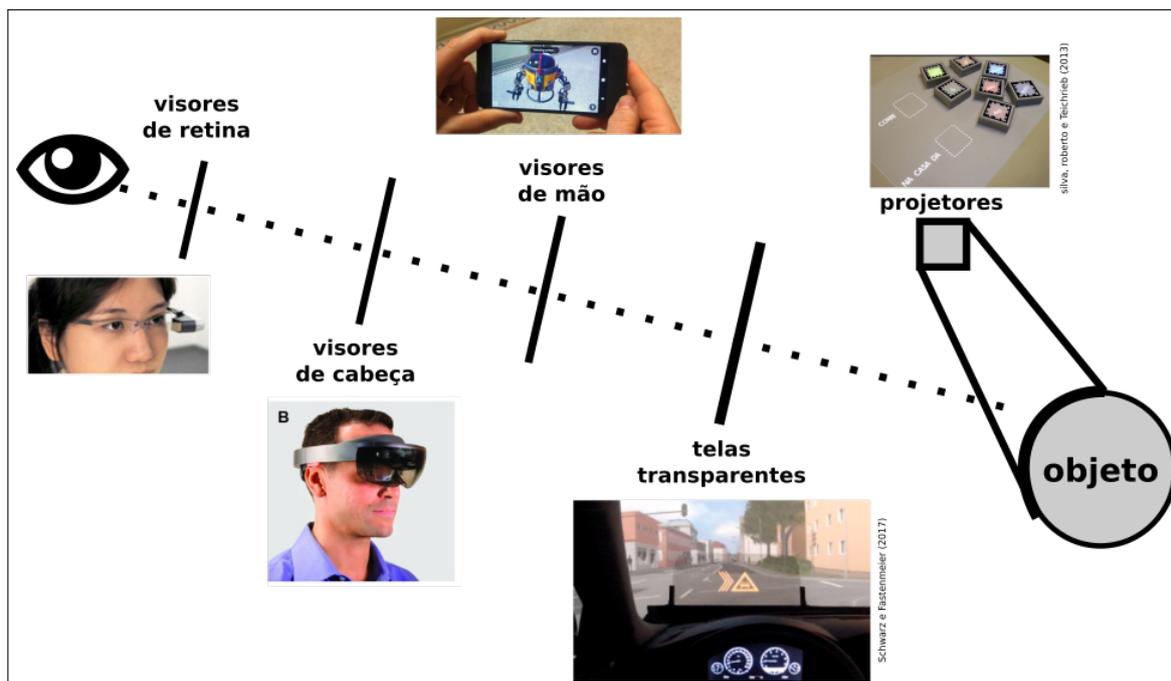


Figura 16. Interseção de objetos virtuais entre o olho humano e o mundo real.

Fonte: Adaptado de Oliver e Raskar (2015).

ao usuário por meio do visor, que adiciona os objetos virtuais na cena. Azuma et al. (2001) cita um terceiro display de cabeça que projeta as imagens diretamente na retina humana usando *lasers* de baixa potência.

Handheld displays

Os *handheld displays* (visores de mão) representam o modo mais comum e acessível de RA. É o que Milgram et al. (1994) denomina *window-on-the-world*, ou seja, uma janela no mundo. Os objetos virtuais aparecem em um visor de um dispositivo que o usuário segura com as mãos (um *smartphone* ou *tablet*), que pode ser comparado a uma lupa. Por meio dessa “janela” ou “lupa” o usuário visualiza o mundo real e os objetos virtuais. Por ser o mais comum, há um conjunto de projetos que visam facilitar o desenvolvimento de aplicações de AR para dispositivos móveis, como o ARCore⁶, da Google.

Projection Displays

Os *projection displays* (visores de projeção) projetam a informação virtual diretamente sobre o ambiente físico. No caso mais simples, utiliza-se um ou mais projetores fixos no ambiente. Outra

⁶ <<https://developers.google.com/ar>>

estratégia é utilizar projetores acoplados à cabeça do usuário, mas há a desvantagem do peso do equipamento (AZUMA et al., 2001). Ambos os casos precisam considerar as características da superfície de projeção, pois superfícies irregulares tendem a deformar a imagem projetada.

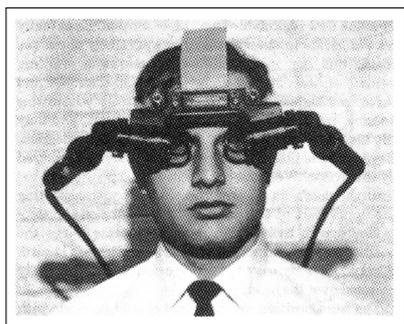
A criação de objetos virtuais que aparecem no mesmo espaço físico que os objetos reais se dá no campo da Realidade Aumentada Espacial (RAE). Quando projetores criam os objetos virtuais, Resch (2016) denomina como Realidade Aumentada Espacial Projetiva (RAEP), aqui tratada simplesmente como RA Projetiva. Ele cita como vantagens da RA Projetiva:

- Não haver erros de profundidade em relação ao ambiente físico, pois os objetos virtuais são projetados diretamente no ambiente;
- Ausência de capacetes especiais ou dispositivos que precise segurar com as mãos, aumentando a segurança e liberando ações manuais;
- O desacoplamento espacial em relação ao usuário permite que os objetos virtuais sejam acessados por grupos maiores de usuários, o que não ocorre com capacetes ou smartphones;
- A resolução obtida com projeção tende a ser maior que a oferecida por dispositivos acoplados à cabeça.

Entretanto, há também desvantagens. Ambientes fortemente iluminados podem ofuscar a imagem projetada, o que exige projetores mais potentes ou projetores a *laser*. O segundo problema é a possibilidade de o usuário obstruir a projeção, colocando as mãos entre o projetor e a superfície que recebe a projeção. O uso de múltiplos projetores sincronizados pode mitigar esse problema, porém aumenta custo e dificulta a instalação.

2.4.1.2 Tecnologias de Rastreamento

O rastreo do usuário, dos objetos virtuais e dos objetos reais, é um aspecto crucial para criar aplicações de RA convincentes (BIMBER; RASKAR, 2005). Um conjunto de métodos são utilizados para esse rastreamento, como uso de sensores como giroscópios, acelerômetros e sensores magnéticos, e também capturação de imagem (RESCH, 2016). Duas técnicas de rastreamento com capturação de imagem são apresentadas a seguir.



(a) Visor de cabeça criado por Ivan Sutherland em 1968.

Fonte: Sutherland (1968).



(b) Visor de mão, em aplicação para smartphone criada com ARCore.

Figura 17. Dispositivos para criar realidade aumentada.

Marcas Fiduciais

As abordagens de rastreamento com base em captação de imagem precisam extrair características presentes na cena observada. Essas características podem ser naturais ou implantadas artificialmente em pontos de interesse. As marcas fiduciais são exemplos de características artificiais criadas para ser facilmente identificáveis por algoritmos de visão computacional. Essas marcas, apresentadas na Subseção 2.3.4, além de serem utilizadas em aplicações com interfaces tangíveis, são também aplicadas no campo da RA.

O processo de identificação das marcas artificiais inicia com a captura da imagem. A esta imagem é aplicado um filtro para transformá-la em preto e branco. Segundo Resch (2016) essa etapa é complexa devido às variações de luminosidade do ambiente. Depois deste filtro, são extraídos os contornos das marcas e por fim o código binário de cores interno à cada marca é decodificado.

No campo da RA projetiva, uma desvantagem do uso de marcas fiduciais é o fato da projeção alterar as características da cena observada, principalmente as cores da marca fiducial, prejudicando sua identificação. Para evitar esse problema é necessário desativar a projeção enquanto a câmera capta as marcas, mas isso impede a projeção em tempo real (RESCH, 2016).

Características Naturais

As características naturais são estruturas que pertencem à cena, sem modificações artificiais. Assim como no caso das marcas fiduciais, os algoritmos de visão computacional precisam identificar objetos de interesse em posições e condições luminosas diversas. Para isso procuram basear-se em

bordas e pontos.

A identificação de bordas foi um dos primeiros métodos de rastreamento de características naturais utilizados, pela facilidade de identificação e robustez em condições variantes de iluminação. Nessa técnica, uma câmera capta um quadro da cena, comparado com um modelo 3D do objeto a ser identificado. Essa comparação tem como saída a estimativa da posição atual do objeto (RESCH, 2016).

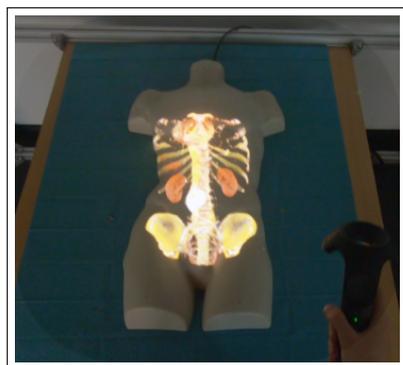
Por outro lado, a identificação de pontos é uma alternativa que também é robusta a variações de luminosidade. Pontos de interesse são previamente cadastrados em uma base de dados criada a partir de um modelo 3D observado de diferentes pontos de vista. Os mesmos pontos são, posteriormente, extraídos da imagem captada pela câmera, e comparados aos pontos armazenados. Essa busca permite então encontrar a posição atual aproximada do objeto real.

2.4.2 RA Projetiva

A RA projetiva é um subtipo da Realidade Aumentada Espacial, a qual normalmente associa um ou mais projetores para produzir o conteúdo virtual e câmeras para captar os objetos reais. Bimber e Raskar (2005) comentam que os ambientes com uso de projeção se popularizaram na década de 1990. Um exemplo desta época é o CAVE, um quarto cujas paredes serviam de telas para criar uma experiência imersiva. Outros exemplos, com menos imersão, tem projeção em áreas menores, como mesas, paredes, paredes curvadas e esferas.

A RA projetiva tem encontrado aplicações em áreas diversas. Uma das abordagens é simular a visualização de estruturas internas a uma superfície, para facilitar o manuseio das mesmas sem a necessidade de abrir a superfície. Exemplos são órgãos humanos, no campo da medicina, e peças de veículos, na mecânica. Em um trabalho neste sentido, Bornemann et al. (2020) combinam três projetores apontados para um manequim e projetam os órgãos humanos permitindo estudar sua localização (Figura 18a). Essa projeção se modifica conforme os movimentos da cabeça do observador, dando às estruturas projetadas um aspecto tridimensional. Essa técnica potencialmente facilitaria cirurgias não invasivas, nas quais atualmente o cirurgião divide a atenção entre o corpo do paciente e um monitor. A projeção dos órgãos sobre o corpo do paciente evitaria a divisão de atenção, mantendo os olhos direcionados para um único local.

Outra aplicação na área médica é o tratamento de pacientes com fobias de pequenos animais. Wrzesien et al. (2015) usam um projetor para mostrar os animais que o paciente teme e ele pode interagir com os mesmos de forma controlada (Figura 18b). No campo da indústria, Sand et al.



(a) Projeção de órgãos internos.
Fonte: Bornemann et al. (2020).



(b) Tratamento de fobias.
Fonte: Wrzesien et al. (2015).



(c) Auxílio em linhas de montagem.
Fonte: Sand et al. (2016).

Figura 18. Aplicações de RA Projetiva.

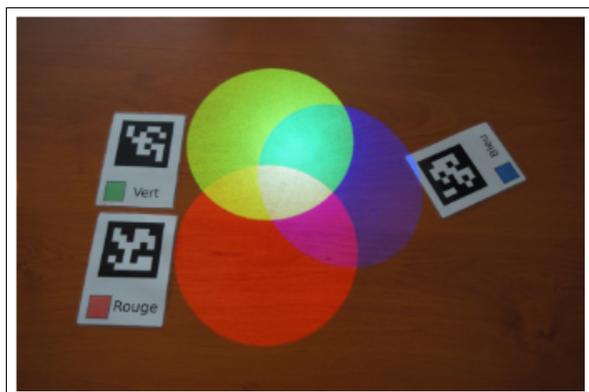
(2016) desenvolveram o smARt.assembly, um sistema de assistência a trabalhadores responsáveis por montagens manuais. O sistema tem um projetor apontado para uma estante com peças, onde destaca a próxima peça a ser utilizada e também mostra como encaixá-la no produto a ser montado (Figura 18c).

2.4.2.1 RA Projetiva Aplicada a Ferramentas Educacionais

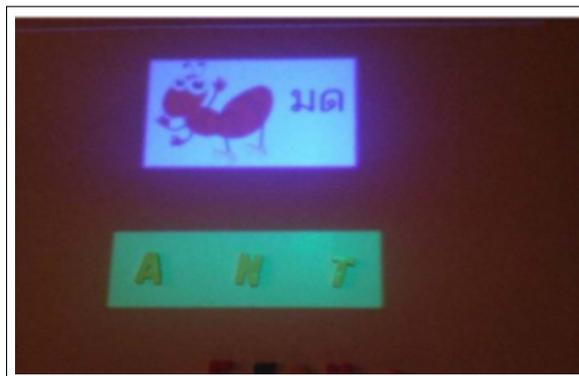
A RA tem potencial educacional ao motivar os estudantes e promover a interação entre o conteúdo e o aluno (SILVA; ROBERTO; TEICHRIEB, 2013). Portanto, além das aplicações na medicina e na indústria, a RA projetiva também se aplica a ferramentas educacionais. Além da RA projetiva, essas ferramentas normalmente utilizam elementos tangíveis que servem como entradas para sistemas câmera-projetor. Câmeras captam os elementos tangíveis e algoritmos de visão computacional identificam esses elementos analisando suas marcas especiais ou suas características naturais. Como saída, um projetor emite os elementos virtuais.

A união entre RA projetiva e interfaces tangíveis em aplicações educacionais favorece interações sociais, o que favorece o aprendizado por meio do *scaffolding* (Subseção 2.1.2). A combinação *RA projetiva-interfaces tangíveis* favorece as interações em grupo de dois modos. Primeiro, a RA projetiva mostra os objetos virtuais a mais de um usuário simultaneamente. Segundo, as interfaces tangíveis permitem a manipulação de objetos físicos por mais de um usuário (BURLESON et al., 2018; HORN; CROUSER; BERS, 2012).

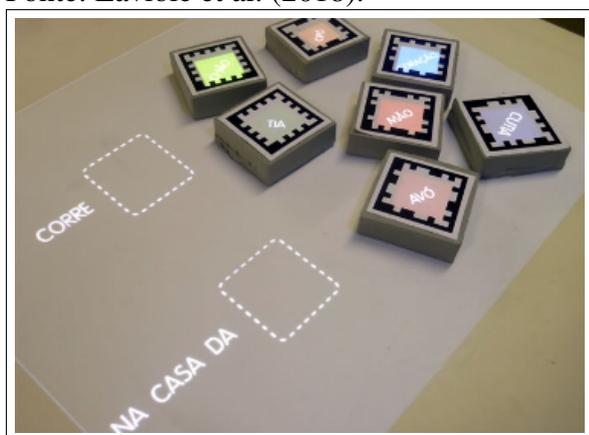
Laviole et al. (2018) apresenta uma ferramenta para exploração de modelos de cores aditivas e subtrativas (Figura 19a). A soma das cores primárias no modo aditivo (verde, vermelho e azul) resulta em branco, e a soma das cores primárias no modo subtrativo (ciano, magenta e amarelo) resulta em



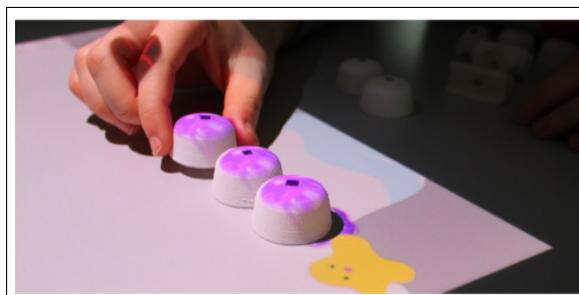
(a) Nectar - Exploração de modos de cores.
Fonte: Laviolle et al. (2018).



(b) Tangible Word Game - Alfabetização.
Fonte: Pinkaew e Waranusast (2014).



(c) ARBlocks - Atividade de alfabetização.
Fonte: Silva, Roberto e Teichrieb (2013).



(d) MaR-T - Atividade de matemática.
Fonte: Beşevli et al. (2019).

Figura 19. Ferramentas educacionais usando RA projetiva.

preto. Na ferramenta de Laviolle, cartas com marcas fiduciais representam as cores, e ao lado de cada carta é projetado um círculo colorido que pode ser sobreposto a outros círculos. Além disso, há uma carta que permite trocar entre o modo aditivo e o subtrativo. Um questionário guia os estudantes na exploração das cartas, com perguntas como: o que é “a cor branca”? O que você vê ao combinar vermelho, verde e azul? Tentando responder às perguntas, os estudantes exploram as combinações de modo ativo e colaborativo, possivelmente compreendendo esse fenômeno físico complexo.

Outras duas ferramentas visam promover a alfabetização. Pinkaew e Waranusast (2014) identificam sequências de letras de plástico e projetam figuras referentes às palavras formadas. Uma câmera de infravermelho capta as letras, técnica que apresentou problemas de diferenciação de letras similares como Q, O, e C, devido a fontes de luz inadequadas. A segunda ferramenta é o ARBlocks, desenvolvido por Roberto et al. (2013). O ARBlocks possui blocos tangíveis customizáveis via projeção de conteúdo virtual, bem como feedbacks luminosos e sonoros. A atividade proposta consistiu na formação de

rimas ao completar frases com blocos tangíveis (Figura 19c). A avaliação com grupo de teste e de controle concluiu que o grupo de teste, com dificuldade de aprendizado, conseguiu aprender de modo similar ao nos assuntos abordados usando RA. A professora da turma também mencionou motivação e engajamento das crianças no uso da ferramenta, a qual afirma tê-las incentivado a lerem mais.

Por fim, a MaR-T é uma ferramenta para auxiliar crianças de 3 a 5 anos no desenvolvimento da representação numérica não-simbólica, ou seja, a comparação entre grandezas (tamanhos, quantidades) antes do reconhecimento dos símbolos numéricos. Desenvolvida por Beşevli et al. (2019), mostra uma sequência de atividades nas quais a criança agrupa pequenas pedras de modo a permitir a passagem de um personagem virtual sobre buracos e rios. As pedras são dispostas vertical ou horizontalmente, ou ainda agrupadas para a criança compreender a conservação de quantidades em diferentes disposições. Em cada etapa, a criança coloca a mão sobre o grupo que possui maior ou menor quantidade de pedras. A aplicação utiliza um projetor acoplado a um *smartphone* Android, e as peças utilizam adesivos refletores que facilitam a identificação por meio da biblioteca OpenCV.

2.4.2.2 Métodos de Entrada

Os métodos de entrada desses sistemas são diversos. No caso do smaArt.assembly o trabalhador controla o sistema usando pedais, para não perder tempo de trabalho manual. Já o sistema de Bornemann et al. (2020) combina a captação de movimentos da cabeça do usuário através de controles remotos que tem sua posição rastreada por sistemas específicos. Ambos os trabalhos citam o reconhecimento de gestos como um método de entrada alternativo a ser testado futuramente. Já o trabalho de Wrzesien et al. (2015) reconhece gestos captados por uma câmera, de modo que o paciente pode interagir com os animais projetados colocando a mão sobre eles. Métodos de entrada tradicionais também são aplicados. Plecher et al (2020) usam RA projetiva em uma exposição de museu, onde os usuários interagem usando *tablet* para pintar uma estátua em exposição, de modo que as cores selecionadas no aparelho são projetadas na escultura. Por fim, as ferramentas educacionais apresentadas (Subseção 2.4.2.1) tem em comum o uso de interfaces tangíveis com ou sem marcas fiduciais, identificadas por câmeras.

2.5 CONSIDERAÇÕES

Este capítulo abordou fundamentos para guiar a criação de interfaces de programação para crianças. Olhar para o indivíduo que usará uma ferramenta, pensando no seu desenvolvimento cognitivo,

limitações motoras e aspectos emocionais é a direção apontada por Montessori, Piaget, Bruner e Papert.

A segunda seção (Seção 2.2) abordou a definição de Pensamento Computacional. Apesar da falta de uma definição estabelecida, percebe-se que há relação com programação, mas que também vai além desta. Os pilares de decomposição, reconhecimento de padrões e abstração podem ser aplicados em resolução de problemas por vezes sem o uso de dispositivos computacionais. A depuração, por estar associada à tarefa de programar, por vezes tem importância secundária e não é mencionada em definições do PC.

A seguir foram abordadas duas classificações de brinquedos programáveis e diferentes alternativas tecnológicas empregadas em suas interfaces. Percebe-se o incentivo ao uso de interfaces tangíveis para crianças, e que a programação em blocos é a metáfora mais utilizada para representar comandos. A preocupação com o design dos blocos também é aparente, por exemplo, ao evitar usar letras em blocos destinados a crianças não alfabetizadas.

Um último item abordado foram novas formas de interação: interfaces de realidade virtual, aumentada e mista. Como demonstrado no próximo capítulo, poucos brinquedos programáveis utilizam interfaces de RA. Por isso, investigar a adequação dessas tecnologias para o público infantil é um tema a ser explorado.

3 TRABALHOS RELACIONADOS

Este capítulo apresenta trabalhos similares ao que se propõe nesta pesquisa. Ele está dividido em três seções. A Seção 3.1 apresenta ferramentas que buscam auxiliar crianças nos primeiros contatos com algoritmos por meio da depuração. A Seção 3.2 apresenta uma revisão sistemática da literatura (RSL) sobre pesquisas que observam a interação de crianças de 3 a 6 anos com BPs. O objetivo desta revisão é compreender quais interfaces são comumente utilizadas, os designs experimentais aplicados e resultados obtidos, para fundamentar a escolha do design experimental desta pesquisa. Por fim, a Seção 3.3 apresenta um mapeamento industrial¹ sobre as interfaces de BPs, que demonstra quais categorias de interfaces estão disponíveis no comércio eletrônico.

3.1 INTERFACES DE BRINQUEDOS PROGRAMÁVEIS

Esta seção demonstra interfaces destinadas para crianças terem os primeiros contatos com algoritmos utilizando BPs. Para isso, identifica as características de cada trabalho e compara com a solução proposta. Os critérios utilizados para a escolha dos trabalhos relacionados foram: a) o público-alvo (crianças a partir dos 4 anos), b) a visibilidade dos algoritmos através de interfaces tangíveis e c) o uso de BPs. Assim, foram analisados quatro trabalhos:

1. Robo-Blocks: uma interface tangível eletrônica para programação de BPs, com foco em ensino de depuração (SIPITAKIAT; NUSEN, 2012).
2. Cubetto: um BP com interface tangível de painel, que indica a execução de cada comando por luzes (<<https://www.primotoys.com>>).
3. Bee-Bot com mesa interativa: uma mesa interativa com objetos tangíveis, que projeta um mapa interativo que reage ao posicionamento dos objetos físicos em sua superfície (BERAZA; PINA; DEMO, 2010).
4. ALERT: interface de programação interativa, em que o BP interpreta comandos à medida que os detecta via câmera (BURLESON et al., 2018).

¹ Método de busca baseado em fontes não acadêmicas, mas que segue um método sistemático. Também denominado mapeamento industrial sistemático.

3.1.1 Robo-Blocks

O Robo-Blocks, de Sipitakiat e Nusen (2012), é um sistema que permite que crianças comandem um BP por meio do encaixe de blocos eletrônicos. Os quatro movimentos básicos do robô são iguais aos do brinquedo RoPE (mover-se para frente, para trás, girar à esquerda e girar à direita), porém as crianças também podem parametrizar a extensão do movimento do robô ajustando um controle giratório presente em cada bloco (Figura 20). Os blocos são encaixados por conectores magnéticos, formando uma sequência. Essa sequência se liga a um bloco principal, que envia os comandos ao BP por comunicação sem fio. O BP é inspirado na tartaruga do ambiente Logo. Ele tem uma caneta acoplada que permite desenhar ao serem utilizados blocos de "pen up" (erguer caneta) e "pen down" (baixar caneta).

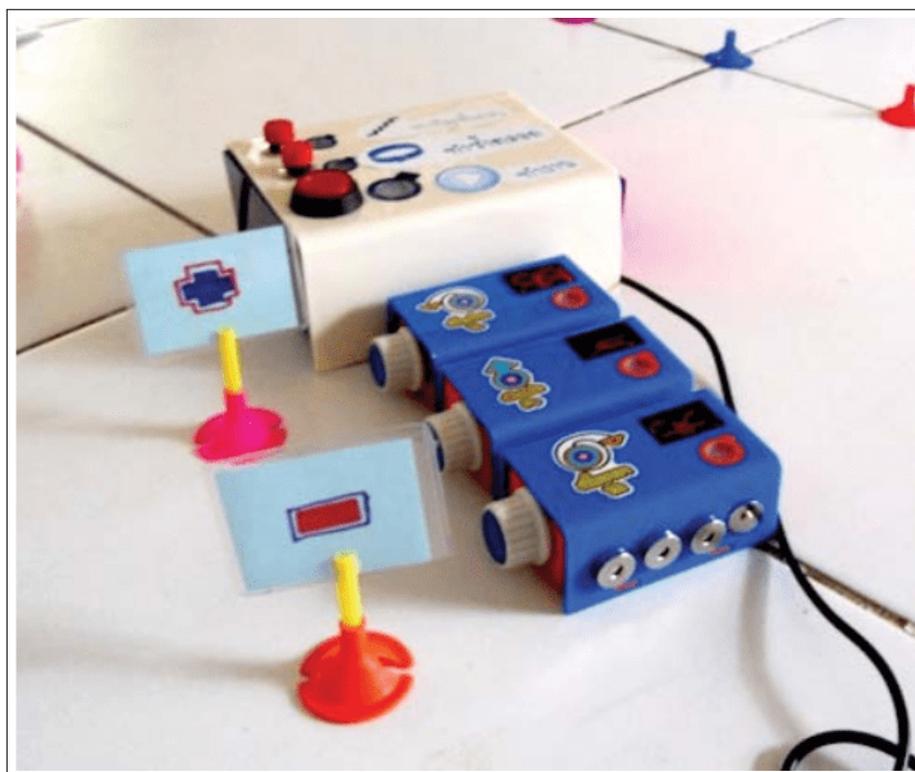


Figura 20. Robo-Blocks.

O Robo-Blocks foi inicialmente desenvolvido com uma interface tangível sem atenção ao processo de depuração. Entretanto, os pesquisadores perceberam que as crianças costumavam direcionar sua atenção para os movimentos do robô, sem prestar atenção nos blocos, o que dificultava encontrar erros. Para resolver este problema os autores criaram a função de execução passo a passo, que permite gerenciar a execução e encontrar erros com mais facilidade.

Além da execução passo a passo, os autores utilizam "objetos passivos", ou seja, que não

influenciam na execução do robô. No caso foi utilizado pequenas bandeiras com as quais a criança identifica os blocos que podem ser a causa de erros. As bandeiras têm ícones de "mais" e "menos". Deste modo, a criança pode colocar a bandeira com sinal positivo ao lado do bloco para indicar que o robô deve girar mais ou mover-se mais para frente, por exemplo.

Outro objeto utilizado para auxiliar na depuração foi o protractor. Destinado a solucionar dúvidas com relação à ângulos, é um simples pedaço de papel com um furo no centro, por onde o robô passa. Quando o robô está parado, a criança coloca o protractor na mesa, sobre o robô. Ao redor do furo há marcações de ângulos, de modo que a criança consegue estimar o quanto o robô precisa se mover relativo à sua direção atual.

Testes com o Robo-Blocks ocorreram com crianças de 5 a 12 anos, porém o estudo formal focou em crianças de 8 a 9. Esse estudo consistiu em atividades de programar o robô para desenhar a primeira letra do nome e andar em um mapa. Os autores identificaram que as crianças tiveram dificuldades em concluir as atividades e recorreram às funções de depuração. Um problema estava no fato de a criança não conseguir observar a cadeia de comandos e robô simultaneamente. Neste sentido, a execução passo a passo foi o modo preferido pelas crianças. O uso das bandeiras para marcar os erros também foi comum, porém os autores ressaltam que a presença de um erro no início da cadeia de blocos interfere em todos os demais movimentos do robô, de modo que parecem também estar errados. Essa característica levava as crianças a marcar todos os demais blocos como errados. Neste caso, os pesquisadores precisaram intervir corrigindo a posição do robô manualmente.

3.1.2 Cubetto

O Cubetto é um brinquedo programável feito de madeira e destinado a crianças entre 4 a 8 anos, que foi desenvolvido pela empresa PrimoToys² a partir de 2013. Ele se inspira no ambiente LOGO, pois se move no chão como a tartaruga robótica desenvolvida de Papert. Além disso, o seu design é inspirado no método Montessori, que incentiva o aprendizado autônomo por meio da interação com materiais concretos, lúdicos.

Os algoritmos que definem os movimentos do Cubetto são feitos com blocos coloridos encaixados em um painel. Há 7 tipos de blocos, sendo quatro blocos direcionais e três blocos lógicos. Os blocos direcionais são frente (verde), trás (roxo), giro à esquerda (amarelo) e giro à direita (vermelho). Os blocos lógicos são movimento aleatório (preto), negação (marrom) e função (azul). O bloco de

² <<https://www.primotoys.com>>

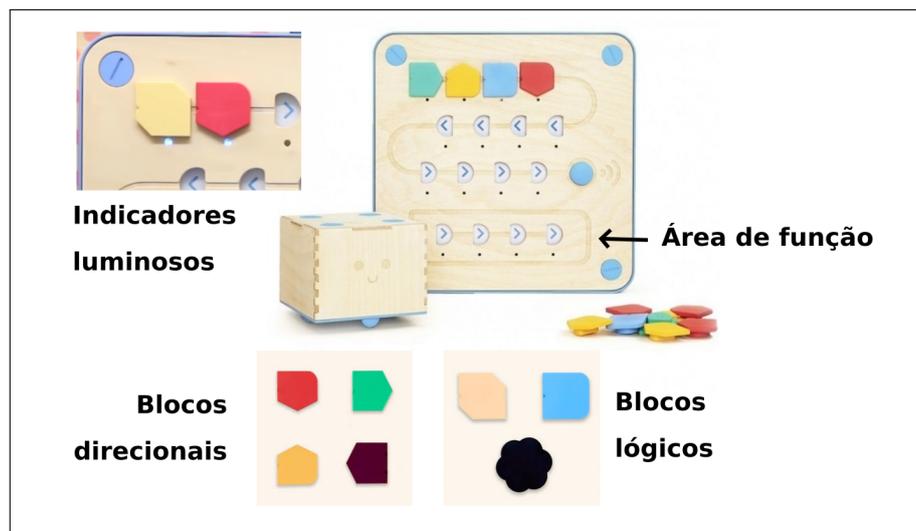


Figura 21. Cubetto: indicadores luminosos, área de função e blocos.

movimento aleatório sorteia um movimento direcional, possibilitando trajetórias diferentes com um mesmo algoritmo. O bloco de negação é associado a um bloco de movimento, modificando-o para movimento contrário. Por exemplo, se o próximo bloco é girar para a esquerda, então o brinquedo gira para a direita. Por fim, o bloco de função faz com que os blocos encaixados em uma área inferior do painel (área de função) sejam executados. Esses blocos especiais, portanto, permitem estender o comportamento além de sequências de comandos. Um bloco de função posicionado na área de função, por exemplo, cria uma função recursiva. Se esta função tiver um bloco de movimento aleatório, o brinquedo executará movimentos diferentes indefinidamente.

O painel do Cubetto possui 16 reentrâncias para encaixe dos blocos, sendo 12 reservadas para o algoritmo principal e 4 dedicadas à área de função. Para cada reentrância há um indicador luminoso que acende quando o bloco é encaixado e pisca quando o bloco é executado. Essa indicação luminosa mostra a sequência de execução de cada bloco, o que é particularmente útil para os blocos de negação e de função. Como o bloco de negação tem sua ação associada ao bloco seguinte (inverte a ação do bloco seguinte), ambos piscam. O mesmo ocorre com o bloco de função, que pisca enquanto os blocos na área de função são executados.

Neste sentido, o design do Cubetto torna visível as instruções do algoritmo e destaca cada passo em execução. Os blocos e o *feedback* luminoso fornecido ao encaixar um bloco no painel favorecem a visibilidade. A execução do algoritmo aparece na interface quando luzes piscam embaixo de cada bloco. Essa visibilidade facilita a interação das crianças e a formação do modelo mental sobre como as peças se relacionam e modificam o comportamento do brinquedo. Os movimentos lentos do brinquedo, associados à indicação dos blocos em execução, promovem a compreensão do algoritmo e facilitam a

depuração³.

3.1.3 Bee-Bot com mesa interativa

Beraza, Pina e Demo (2010) criaram uma mesa interativa para ser utilizada com a Bee-Bot por crianças de 4-5 a 12-14 anos. Essa mesa tem superfície semitransparente, na qual um projetor cria mapas virtuais interativos, projetando-os por baixo. A imagem projetada muda de acordo com os movimentos da Bee-Bot e acompanha a manipulação de outros elementos tangíveis. Por exemplo, quando a Bee-Bot passa de um quadrado para outro do mapa projetado, este quadrado é destacado.

A projeção de elementos virtuais de acordo com os elementos tangíveis ocorre devido à marcas fiduciais da biblioteca reactIVision coladas na parte inferior desses elementos. Como a superfície da mesa é semitransparente, uma câmera embaixo da mesa consegue captar as marcas fiduciais. As imagens captadas são analisadas por um software que controla as projeções.

Além da mesa, o trabalho propõe a criação de um software de apoio aos professores para que criem e configurem as atividades com o brinquedo programável. Esse software busca atender aos seguintes requisitos:

- Ter tapetes ou cenários adaptados para cada atividade;
- Poder interromper uma atividade e continuar depois;
- Registrar o progresso das crianças em determinado problema e acompanhar sua evolução no tempo;

Uma atividade de demonstração permite entender as potencialidades da tecnologia (Figura 22). Nessa atividade, o usuário pode decidir o mapa a ser utilizado; decidir as posições de início e fim do trajeto a ser percorrido; e ver o trajeto percorrido com o número de passos executados. A visualização do trajeto percorrido permite à criança refletir sobre os comandos inseridos e depurar possíveis erros.

³ Prêmios de design recebidos pelo Cubetto:

- London Design Award (2016) - <<https://drivenxdesign.com/LON16/project.asp?ID=15112>>
- German Design Award (2017) - <<https://www.german-design-award.com/en/the-winners/gallery/detail/9304-cubetto.html>>
- RedDot Award (2016) - <<https://www.red-dot.org/project/cubetto-26086>>



Figura 22. Mesa interativa com a Bee-Bot.

Fonte: Beraza, Pina e Demo (2010)

Nessa demonstração, cada quadrado do mapa contém uma forma geométrica de cor e tamanho variado. Isso possibilita que a criança exercite a classificação e abstração. A classificação seria a habilidade de identificar propriedades ou categorias, e relacionar essas categorias ou classes entre si. Nessa atividade, a criança classifica formas geométricas por propriedades como cor, formato e tamanho. Assim, a criança aprende a abstrair e observar apenas as propriedades relevantes para formar uma categoria. Esse mesmo tipo de tapete com formas geométricas é utilizado pelo brinquedo RoPE, porém em seu formato físico.

3.1.4 ALERT

O ALERT (BURLESON et al., 2018), assim como os trabalhos anteriores, é uma interface de programação de brinquedos programáveis. O seu diferencial é que o robô interpreta os comandos à medida que ficam acessíveis no seu campo de visão. Cada robô possui uma câmera, que detecta novos comandos quando o robô se move. Por exemplo, se o robô anda para frente e capta o comando “vire 90° à direita”, ele interpreta e executa esta ação. Nisto percebe-se o princípio do feedback imediato defendido por Norman (1990). A execução do comando imediatamente ao ser detectado pelo brinquedo explicita a relação entre comando e resultado.

Para identificar os comandos captados o sistema interpreta marcas fiduciais da biblioteca reactTIVision, assim como o trabalho de Beraza, Pina e Demo (2010). Essas marcas podem estar

fixadas em folhas impressas distribuídas pelo chão, coladas em outros robôs ou também ser projetadas na superfície.

A vantagem em relação à inserção de comandos por meio de botões é a variedade de comandos possível de desenvolver sem a necessidade de um botão físico. Pode haver, por exemplo, uma marca representando o comando “vire 180º” ou “toque um som” sem necessidade de alterações de hardware. A Figura 23 demonstra o uso do ALERT. As marcas fiduciais aparecem no cenário partindo de um projetor e representam, segundo o autor, comida e as bordas do cenário. O personagem da direita também tem marcas fiduciais e isso permite ao personagem da esquerda identificá-lo.

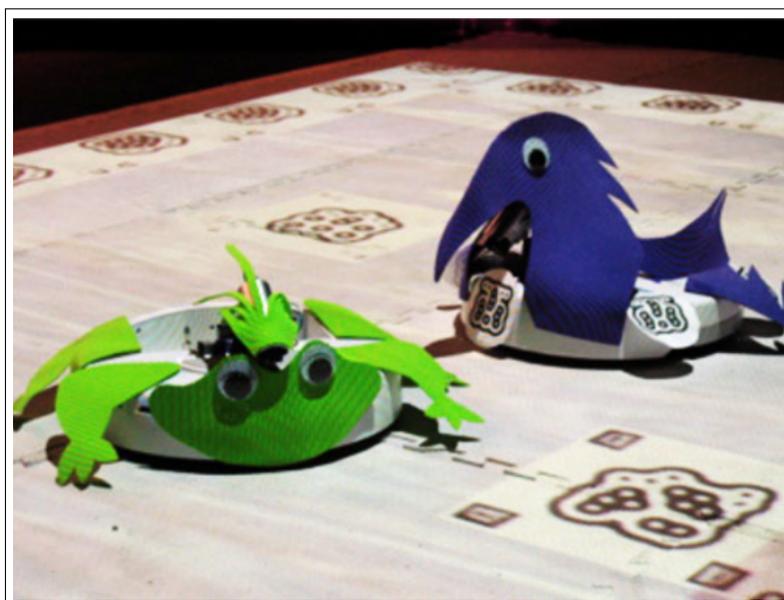


Figura 23. ALERT.

Fonte: Burleson et al. (2018)

Com relação à depuração, a programação interativa se apresenta como um diferencial. A execução dos comandos à medida que o brinquedo os detecta, possibilita às crianças corrigirem o algoritmo durante a execução. Os autores exemplificam que se o brinquedo faz um movimento em U, quando o esperado era um giro à direita, as crianças podem adicionar novos comandos na frente dele. Deste modo a criança tem interação constante tanto com o robô quanto com o algoritmo, que se torna algo móvel.

3.1.5 Análise Comparativa

As seções anteriores apresentaram quatro trabalhos relacionados com programação de brinquedos programáveis.. Esta seção compara estes trabalhos quanto às seguintes características: idade

do público-alvo, tipo de interface, e depuração (Tabela 4). Por fim, os trabalhos são posicionados em relação à solução proposta.

Em relação ao público-alvo, o Robo-Blocks foca em crianças de 8 a 9 anos, pois a criança precisa entender basicamente o significado dos algarismos numéricos ao parametrizar o ângulo do giro do robô, e nesta idade as crianças normalmente já aprenderam a ler (SNOW; BURNS; GRIFFIN, 1998). Em contrapartida, o Cubetto, a Bee-Bot⁴ e o ALERT não dependem diretamente de números, e atendem um público a partir dos 4 anos.

Com relação à interface, todos os trabalhos utilizam interface tangível, mas seguindo diferentes abordagens. O Robo-Blocks tem blocos eletrônicos encaixados entre si e que possuem um display que mostra o tamanho do avanço ou giro. A programação do Cubetto também se dá por blocos tangíveis, porém sem visores e controles ajustáveis. Tudo o que a criança faz é encaixar os blocos em um painel. O ALERT tem blocos ainda mais simples. Foram usadas folhas de papel com as marcas fiduciais impressas, o que tende a ser mais barato e fácil de reproduzir. Por fim, a interface tangível da Bee-Bot não tem blocos, mas sim botões (YU; ROQUE, 2019). Apesar das diferentes abordagens, todos os trabalhos mencionam engajamento das crianças durante as interações, sejam elas com blocos eletrônicos, com as folhas de papel ou com os botões.

Por fim, em relação à depuração, o Robo-Blocks é a interface com maior foco em facilitar a atividade de encontrar erros em algoritmos. Isso se dá por duas características: marcadores físicos (bandeiras) para indicar possíveis erros, e a função de execução passo a passo. No ALERT, os elementos de execução passo a passo são menos necessários. O brinquedo se move lentamente, e com isso há mais tempo para a criança olhar os comandos e comparar com os movimentos do robô. Além disso, o algoritmo pode ser alterado durante a execução. Se um comando desvia o brinquedo da rota, os passos subsequentes podem ser alterados para reajustá-la. A associação entre comando e ação também é mais óbvia no ALERT, pois o brinquedo precisa se aproximar do comando a ser executado. Quando um erro ocorre, fica claro qual foi o comando responsável. Quando os blocos eletrônicos ficam distantes do brinquedo esse mapeamento entre comando e ação é menos óbvio. Segundo Sipitakiat e Nusén (2012) as funções de depuração foram uma tentativa de resolver esse problema de mapeamento. A depuração no Cubetto é similar ao Robo-Blocks, pois o painel de programação fica distante do robô. Além disso, o Cubetto não tem execução passo a passo. Por fim, a interface de RA utilizada com a Bee-Bot (BERAZA; PINA; DEMO, 2010) não tem visibilidade de código e nem funções de execução passo a passo, mas é similar ao trabalho aqui apresentado devido às tecnologias empregadas.

⁴ Existem versões com texto nos botões clear, go e pause, mas os botões provocam sempre o mesmo comportamento.

Quadro 4. Análise comparativa.

Projeto ou ferramenta	Idade do público-alvo	Tipo de interface	Depuração
Robo-Blocks	8 a 9 anos	Tangível com blocos eletrônicos	Execução passo a passo, protractor e marcadores de erros
Cubetto	4 a 8 anos	Tangível com painel eletrônico	Destaca os blocos em execução
Bee-Bot com mesa interativa	4-5 a 12-14 anos	Tangível com botões	-
ALERT	6 anos	Tangível com projeção	Permite alterar o comportamento do robô durante a execução
RoPE AR	4 a 6 anos	Tangível com projeção	Destaca os blocos em execução

Fonte: O autor.

Os trabalhos apresentados usam diferentes abordagens visando o mesmo objetivo: facilitar os primeiros contatos de crianças com algoritmos. Os quatro trabalhos aplicam interfaces tangíveis, mas de diferentes formas: há o uso de painel, blocos e botões. Assim como estes, o RoPE AR também usa interface tangível. A diferença está na união do tangível com a realidade aumentada. O trabalho de Beraza, Pina e Demo (2010) e o ALERT também aplicam RA, mas não há associação entre elementos os virtuais e os blocos que formam o algoritmo. Isso não significa uma desvantagem dos trabalhos citados, mas aponta uma lacuna de investigação a que este trabalho busca preencher.

Outro diferencial do RoPE AR é a tentativa de facilitar a instalação e o transporte. Isso é importante para produtos a serem usados em sala de aula. O trabalho de Beraza, Pina e Demo (2010) necessita de uma mesa com tampa semi transparente e de um notebook. O RoPE AR, por outro lado, utiliza apenas um suporte para o projetor, um projetor pequeno e um smartphone. Por fim, um último diferencial é que no RoPE AR a projeção ocorre por cima, o que permite projetar não apenas figuras numa superfície inferior aos elementos tangíveis, mas também sobre eles.

3.2 REVISÃO SISTEMÁTICA: PESQUISAS DE IHC COM CRIANÇAS

Para compreender como são feitas as pesquisas sobre brinquedos programáveis com crianças, o método utilizado foi o de revisão sistemática (KITCHENHAM; CHARTERS, 2007). Esse método revisa estudos a respeito de uma pergunta de pesquisa, tópico ou fenômeno de interesse. No processo

Mesmo que não saiba ler, a criança consegue mapear a ação do botão com sua funcionalidade.

busca-se identificar, avaliar e interpretar as publicações relevantes para o tema em questão, e evitar o viés de quem produziu a pesquisa. Três questões exploratórias guiaram a busca:

- QE1 — Quais categorias de interfaces são usadas em pesquisas sobre aprendizado de algoritmos na Educação Infantil?
- QE2 — Quais métodos e design de experimentos são utilizados?
- QE3 — O que as crianças aprendem ao interagir com brinquedos programáveis?

A seguinte busca foi adaptada para os artigos publicados nos anos de 2015 a 2020 nos repositórios ACM Digital Library, ERIC, IEEE Xplore e Science Direct:

(crianças ou "jardim de infância") E (atividades OU "práticas pedagógicas") E (robótica ou "brinquedos programáveis") E (pensamento computacional) .

3.2.1 Resultados

Sobre os resultados obtidos com as buscas nos repositórios foram aplicados os critérios de inclusão e exclusão, primeiramente sobre os resumos e em seguida sobre o texto completo (Quadro 5). Com isso restaram 7 artigos.

Quadro 5. Critérios de inclusão e exclusão.

Critérios de inclusão	Critérios de exclusão
1. Estudos primários indicando intervenção relacionada a ensino/aprendizado de algoritmos na Educação Infantil 2. Estudos publicados entre 2015 e 2020 3. Público com idade entre 3 e 6 anos	1. Artigos em línguas diferentes de inglês e português 2. Título ou resumo mencionando nível diferente de Educação Infantil 3. Foco em formar professores 4. Menos de 5 páginas 5. Estudos secundários ou terciários 6. Público-alvo diferente desta pesquisa 7. Publicações duplicadas 8. Estudo que não aborda criação de algoritmos por crianças 9. Estudo cujo texto completo não foi possível acessar

O critério de inclusão 3 (idade entre 3 a 6 anos) filtrou 82 trabalhos com um público-alvo de crianças maiores de 6 anos. O segundo critério que filtrou mais artigos (CI1) filtrou 30 artigos que não

apresentaram nenhuma intervenção com crianças para mensurar resultados de alguma ferramenta ou aprendizado de algoritmos. Outros 15 trabalhos focaram em públicos específicos que se diferenciam do público desta pesquisa. O Quadro 6 apresenta os artigos que passaram no filtro.

Tabela 2. Resultados da RSL

Repositório	Res.	CE1	CE2	CE3	CE4	CE5	CE6	CE7	CE8	CE9
ERIC	4	3	2	2	2	2	2	2	2	2
ACM Digital Library	39	33	11	11	10	7	3	3	2	2
Science Direct	7	7	3	3	3	2	2	2	2	2
IEEE Xplore	114	91	36	32	32	21	13	11	11	1
Filtrados pelo critério	-	30	82	4	1	15	12	2	1	10
Artigos restantes	164	134	52	48	47	32	20	18	17	7

Fonte: O autor.

Quadro 6. Artigos resultantes da RSL

Publicação	Métodos utilizados / interfaces	Resultados / aprendizados
Repiso e González (2019) <i>Robotics to develop computational thinking in early Childhood Education</i>	Intervenção com a Bee-Bot® durante 7 encontros. Pesquisa quantitativa, quasi-experimental, com pré e pós teste. As crianças programaram usando botões e cartas. A avaliação utilizou desafios de programação com histórias lúdicas. As professoras e pesquisadores atribuíram notas de 0 a 5 conforme Bers et al. (2014).	A intervenção causou resultados positivos no grupo experimental nos aspectos de sequenciamento, correspondência ação-instrução e depuração.
Nam, Kim e Lee (2019) <i>Connecting Plans to Action: The Effects of a Card-Coded Robotics Curriculum and Activities on Korean Kindergartners</i>	Intervenção utilizou a TurtleBot durante 8 encontros de 90 minutos com 53 crianças, uma vez por semana. As crianças planejaram o deslocamento desenhando e programaram usando cartões. Pesquisa quantitativa, quasi-experimental, com pré e pós teste. As professoras participaram da intervenção, auxiliando as crianças.	Apresentou resultados positivos nas habilidades de sequenciamento e resolução de problemas para o grupo experimental.
Lieto et al. (2017) <i>Educational Robotics intervention on Executive Functions in preschool children: A pilot study</i>	Intervenção com a Bee-Bot® durante 6 semanas, duas vezes por semana, sendo 75 minutos por sessão. Divisão em grupos de 3 a 4 crianças, cada criança com um robô. Aumento gradual do tamanho do caminho a ser percorrido. Execução de testes de programação e psicológicos (memória, visão espacial e atenção). Uso de pré e pós teste.	Resultados indicaram aumento da capacidade de atenção e memória de trabalho. Melhorias na habilidade de programação das últimas 3 sessões comparadas às 3 primeiras.
Sheehan et al. (2019) <i>Parent-child interaction and children's learning from a coding application</i>	Pesquisa quantitativa. Em dupla com seus pais, 31 crianças interagiram com o ScratchJr durante 10 minutos. Pesquisadores gravaram e transcreveram as conversas entre pais e filhos. Foram extraídas falas sobre linguagem espacial e perguntas. A linguagem foi relacionada com a produção e compreensão de comandos.	Os pais utilizaram mais termos de linguagem espacial e fizeram mais perguntas do que seus filhos, para auxiliá-los. O número de perguntas foi um preditor negativo da produção e compreensão de comandos.
Pila et al. (2019) <i>Learning to code via tablet applications: An evaluation of Daisy the Dinosaur and Kodable as learning tools for young children</i>	Pesquisa qualitativa com pré e pós teste. Vinte e oito crianças de 4 a 6 anos usaram, durante uma semana, 2 aplicativos para ensino de programação (Kodable e Daisy the Dinosaur). A pesquisa captou o quanto as crianças gostaram de cada aplicativo e a influência do gênero no desempenho. Também foi avaliado o reconhecimento de comandos antes e após a intervenção, com notas de 0 a 5 conforme Bers et al. (2014).	Testes t identificaram um aumento significativo no reconhecimento dos comandos do aplicativo Daisy. Após uma semana foi possível identificar a compreensão de habilidades gerais de codificação. O “apelo” do aplicativo teve correlação com o aprendizado resultante, e o gênero não afetou o desempenho.

Publicação	Métodos utilizados / interfaces	Resultados / aprendizados
Burleson et al. (2018) <i>Active Learning Environments with Robotic Tangibles: Children's Physical and Virtual Spatial Programming Experiences</i>	Pesquisa qualitativa comparando o uso de duas ferramentas de programação espacial, sendo uma virtual e outra tangível (Robopad e ALERT). Nove crianças de 6 anos, divididas em 4 grupos, interagiram durante 5 dias (num calendário de 2 semanas) com ambas as ferramentas.	Usando o ALERT as crianças programaram, colaborativamente, sequências para alterar o comportamento do robô. A ferramenta Robopad induziu mais a alterar/depurar o “programa” em tempo de execução.
Heljakka et al. (2019) <i>Gamified Coding: Toy Robots and Playful Learning in Early Education</i>	Pesquisa qualitativa. Durante um período de 6 meses os pesquisadores disponibilizaram os brinquedos programáveis Dash e Botley a duas professoras e 21 crianças de 5 anos. As crianças brincaram livremente com os brinquedos e as professoras registraram o quão rápido as crianças aprenderam a programá-los. Os pesquisadores visitaram a sala de aula e entrevistaram crianças e professoras.	As crianças aprenderam a programar os brinquedos e também criaram brincadeiras, como criar um túnel com as pernas para o Dash passar e construir caminhos para o Botley percorrer.
Esta pesquisa	Pesquisa mista. Coleta automática de interações. Uso do brinquedo RoPE com e sem realidade aumentada.	Pretende-se mensurar a colaboração, a depuração, e o número de erros ligados à lateralidade.

3.2.2 QE1 — Quais categorias de interfaces são usadas em pesquisas sobre aprendizado de algoritmos na Educação Infantil?

A categoria de interface mais citada foram os brinquedos controlados por botões. A Bee-Bot apareceu em duas pesquisas, e outros brinquedos programados por botões foram a TurtleBot e o Botley. Aplicativos citados foram o ScratchJr, a interface do Dash, o Kodable, e *Daisy the Dinossaur*. Outra interface citada é a espacial: Burleson et al. (2018) apresenta o ALERT e o Robopad, em que os agentes se movem no espaço e captam os comandos do ambiente.

Além das interfaces, outra variável a ser considerada são as atividades realizadas. Uma mesma interface, utilizada de modos diferentes, pode gerar diferentes aprendizados. A sequência de atividades do TangibleK Robotics Curriculum, desenvolvida por Bers et al. (2014) foi aplicada por Repiso e González (2019) e por Pila et al. (2019). Dividida em 7 encontros, o primeiro dia de atividades iniciou com a exploração da interface da Bee-Bot, seguida (nos dias 2 e 3) da programação do brinquedo para se deslocar sobre tapetes. Nos dias 4 e 5 foram introduzidas cartas correspondentes às instruções, para serem sequenciadas e depois comparadas com os movimentos do brinquedo. Nos dias 6 e 7 foram mostradas sequências de comandos com erros, e as crianças exercitaram a depuração ao encontrar esses erros.

Lieto et al. (2017) apresenta atividades de dificuldade incremental, em que as crianças precisavam programar a Bee-Bot para se deslocar sobre um tapete de formas cada vez mais complexas. Heljakka et al. (2019) não definiram nenhuma atividade específica para as crianças, apenas permitiram a exploração dos brinquedos Botley e Dash por 6 meses.

3.2.3 QE2 — Quais métodos e design de experimentos são utilizados?

O método mais utilizado foi a pesquisa quasi-experimental, com pré e pós teste. Das 7 pesquisas, 4 foram quantitativas, 2 qualitativas e uma mista. Além disso, Nam, Kim e Lee (2019) e Repiso e González (2019) utilizaram grupos de controle e experimental.

O tempo das intervenções variou consideravelmente. As pesquisas objetivando mensurar efeitos decorrentes das interações com os brinquedos foram mais longas, durando de 7 encontros a 6 meses. A pesquisa que buscou avaliar a usabilidade foi mais rápida, durando 10 minutos.

As amostras utilizadas foram de conveniência, com experimentos realizados nos ambientes escolares. O tamanho das amostras variou entre 9 e 31 crianças, que trabalharam em grupos em 6 das

7 pesquisas. As fontes de dados utilizadas foram vídeos, mapa de eventos, rubricas, transcrições, e entrevistas.

3.2.4 QE3 — O que as crianças aprendem ao interagir com brinquedos programáveis?

Os trabalhos citam principalmente o aprendizado de sequenciamento e de algoritmos em geral. Pila et al. (2019), após a interação de um conjunto de crianças durante 5 dias com os aplicativos Daisy e Kodable, perguntaram às crianças “O que é programar?”. A nota máxima foi dada a respostas que apresentavam a ideia de causa e efeito aplicável à tecnologia em geral, por exemplo “programar é colocar código em algo para fazer alguma ação”. Os resultados indicaram que as crianças não evoluíram na capacidade de expressar verbalmente seu entendimento de codificação.

Outro aspecto avaliado por Repiso e González (2019) e Pila et al. (2019) é a compreensão dos símbolos da interface. Repiso e González (2019) investigaram a compreensão sobre a correspondência entre ação e instrução, que significa mapear qual ação o robô executa em correspondência a um símbolo. A partir das rubricas os pesquisadores identificaram uma evolução significativa no grupo experimental. Pila et al. (2019) mostrou quatro símbolos da interface do aplicativo Daisy para as crianças e solicitou que explicassem verbalmente o significado do símbolo. No pré-teste a média de acerto foi de 0,5 e no pós-teste esse valor subiu para 2,3.

Nam, Kim e Lee (2019) avançaram nessa questão e captaram a compreensão de sequências. O instrumento utilizado solicita que, dada uma imagem, outras 4 imagens com eventos sequenciais sejam organizados. Aplicando essa técnica com crianças de 4 anos, os resultados apontaram evolução significativa na nota de sequenciamento para o grupo experimental.

3.3 MAPEAMENTO INDUSTRIAL: INTERFACES DE BRINQUEDOS PROGRAMÁVEIS

O mapeamento industrial buscou encontrar o maior número possível de brinquedos programáveis que estão ou estiveram disponíveis para comercialização⁵. A busca por estas fontes se justifica por abranger empresas e produtos que não estão diretamente ligados a publicações acadêmicas. Portanto, a pesquisa não utilizou bases científicas, mas sim plataformas de comércio eletrônico e

⁵ O mapeamento fez parte de uma pesquisa voltada a brinquedos com características inteligentes, como IA e IoT. Nesta seção os dados serão avaliados com foco em realidade aumentada.

buscadores. O mapeamento seguiu três etapas (COOPER; DANN; PAUSCH, 2000):

1. Definição das questões de pesquisa
2. Planejamento do processo de busca
3. Definição dos critérios para filtrar os resultados

As próximas subseções detalham essas etapas.

3.3.1 Questões de Pesquisa

Três questões exploratórias guiaram o processo de busca:

1. Quais são as categorias de interfaces mais frequentemente utilizadas em brinquedos programáveis?
2. Como a realidade aumentada é aplicada em brinquedos programáveis?
3. Quais são os conceitos de algoritmos abordados?

3.3.2 Processo de Busca

A segunda etapa define a *string* de busca a ser aplicada em campos de pesquisa. O foco esteve em programação, resultando na seguinte *string*:

((coding OR programmable) AND toys) .

Seguindo a estratégia também apresentada no mapeamento feito por Kitchenham e Charters (2007), uma exploração inicial indicou o uso das seguintes fontes de busca:

- Plataforma de comércio eletrônico Amazon.com
- Postagens de blogs encontrados em pesquisas no Google e no DuckDuckGo
- Plataforma de financiamento coletivo Kickstarter.com
- Vídeos do Youtube

A string de busca foi aplicada nos sites que oferecem campos de pesquisa. Outra fonte de busca foram listas de produtos categorizados pelos sites consultados. O site [kickstarter.com](https://www.kickstarter.com), por exemplo, tem uma área específica sobre projetos de robótica, enquanto a plataforma [amazon.com](https://www.amazon.com) possui uma categoria de produtos denominada *coding toys*. Essas pré-categorizações foram adicionados aos resultados da busca textual. Por fim, a abrangência do mapeamento foi validada por meio da consulta de artigos de revisão de modo a encontrar brinquedos faltantes. Cinco brinquedos foram adicionados nesta etapa.

3.3.3 Filtro

Três critérios guiaram a inclusão dos brinquedos nos resultados. Primeiramente, a presença de alguma forma de inserir comandos a serem executados pelo brinquedo. O segundo critério é a presença de algum componente tangível, ou seja, aplicativos sem interação com brinquedo físico foram descartados. O último critério é possuir algum componente eletrônico, de forma que haja um processamento dos comandos programados para controlar sua execução. Esse critério eliminou os jogos de tabuleiro e livros de programação (HAMILTON et al., 2020).

Após encontrar brinquedos adequados aos critérios de inclusão, a próxima etapa foi obter dados sobre as interfaces de programação. Para cada brinquedo foi executada uma pesquisa por vídeos demonstrando o uso do brinquedo. Outras fontes de dados foram as descrições presentes nas páginas de comércio eletrônico e páginas dos fabricantes.

3.3.4 Resultados

Um total de 86 brinquedos resultaram do processo de busca. Destes, 56 serviram como objeto de análise para responder às três perguntas de pesquisa. Os outros 30 brinquedos não foram analisados, dado que foram adicionados posteriormente aos resultados ou por serem versões similares de algum dos brinquedos analisados. Os resultados estão disponíveis no endereço <<https://bit.ly/35cUbZ1>>. O Quadro 7 sumariza os brinquedos utilizados na análise.

Quadro 7. Brinquedos resultantes do mapeamento industrial.

Nome	Fabricante	Ano	Nome	Fabricante	Ano
Big Trak / Big Track Junior	Milton Bradley	1979	Sphero SPRK+	Sphero	2016
Topobo	MIT Tangible Media Group	2003	Go Robot Mouse	Learning Resources	2016
Finch	BirdBrain Technologies	2010	Coji	WowWee	2016
Bee-Bot	TTS Group	2011	KUMITA	ICON Corp	2016
KIBO	Kinderlab Robotics	2012	RoPE	SmartFun Brasil	2017
Dr. Wagon	Stanford	2012	Harry Potter Coding Kit	KANO	2017
Ollie	Sphero	2012	Meccano-Erector Meccanoid	Spin Master	2017
RoboTami Creative Robot Kit	Robotron	2012	Lego Boost	Lego	2017
Lego Mindistorms EV3	Lego	2013	Anki Cozmo	Digital Dreams	2017
Romo	Romotive	2013	Sam Curious Cars Kit	Sam Labs	2017
Pro-Bot	TTS Group	2014	Airblock	Makeblock	2017
Dot	MakeWonder	2014	Augie	Pai Technology	2017
Dash	MakeWonder	2014	Minion Mip	WowWee	2017
Edison	Edison	2014	Miko 2	Miko	2018
Ozobot	Ozobot	2014	Kids First Coding e Robotics	GIGO	2018
TinkerBots	TinkerBots	2014	Botley	Learning Resources	2018
TiddlyBot	Agilic	2014	Q-Scout	RoboBloq	2018
Guimo	Guimo Toys	2015	InO-Bot	TTS Group	2018
BlueBot	TTS Group	2015	Codey Rocky	Makeblock	2018
Osmo Coding	Tangible Play Inc	2015	Aibo - ERS-1000	Sony	2018
Cubetto	PrimoToys	2015	Root	iRobot	2018
Cue	MakeWonder	2015	Rugged Robot	TTS Group	2019
Vortex	DFRobots	2015	Aukfa / JDBaby / HBuds	Aukfa	2019
mBot	Makeblock	2015	Qobo	Robobloq	2019
Robotiky	Robotiky	2015	mTity	Makeblock	2019
SmartiBot	The Crafty Robot	2015	Botzees	Pai Technology	2019
Robo Wunderkind	Robo Wunderkind	2015	Mojobot	Project Lab	2019
Code a Pillar / Codipédia	Fisher Price - Mattel	2016	Mochi	Mochi	2019

Quarenta e seis brinquedos (85%) foram lançados a partir de 2014, ano seguinte ao aumento no interesse pelo tema do Pensamento Computacional. A Figura 24 demonstra o número de brinquedos lançados por ano, e, comparado com a Figura 25, há crescimento no número de brinquedos lançados a partir de 2013. Portanto, pode haver uma relação entre o interesse pelo Pensamento Computacional e BPs no mesmo período.

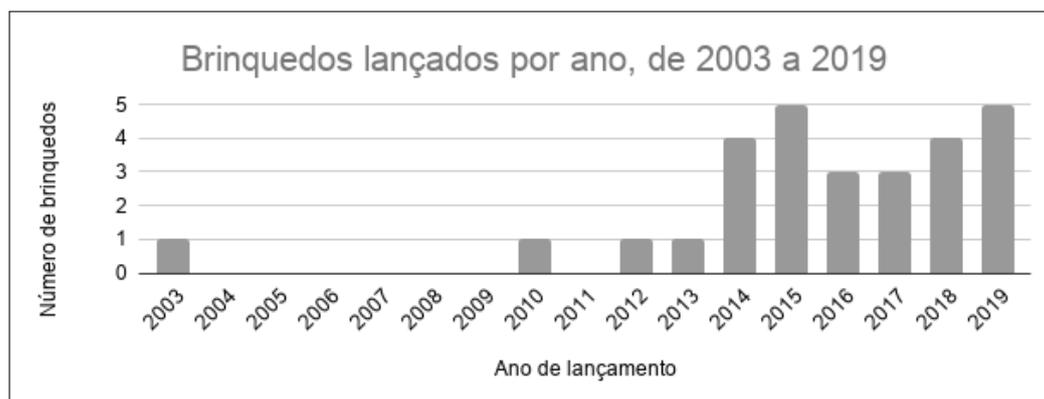


Figura 24. Brinquedos lançados por ano entre 2003 e 2019.

Fonte: O autor.

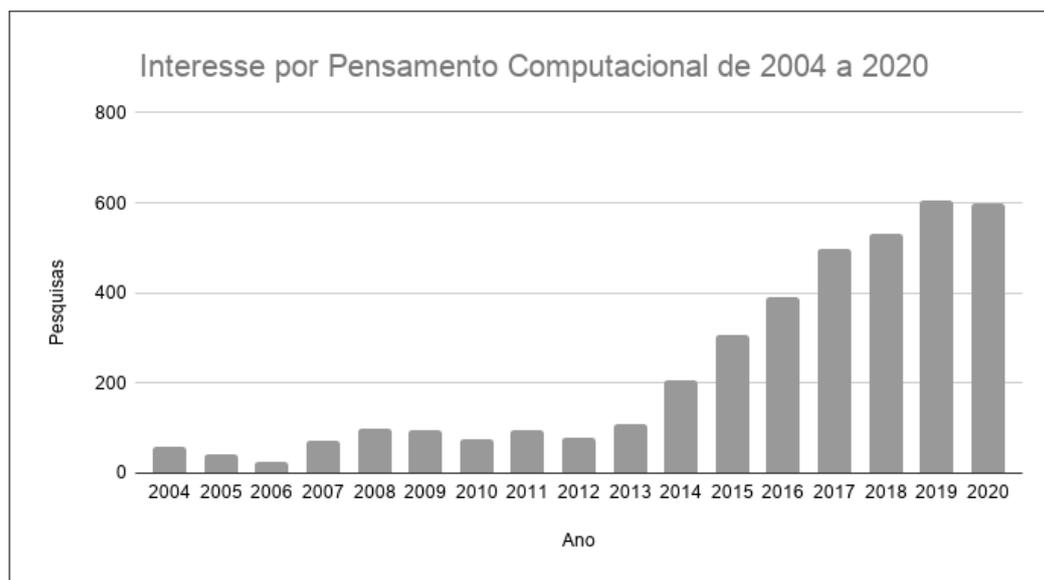


Figura 25. Interesse pelo tema Pensamento Computacional, segundo o Google Trends.

Fonte: O autor.

3.3.5 QE1 - Quais são as categorias de interfaces mais frequentemente utilizados em brinquedos programáveis?

A resposta à pergunta *QE1 - Quais são as categorias de interfaces mais frequentemente utilizados em brinquedos programáveis?* depende da perspectiva de análise dessas interfaces. Uma

pesquisa pode observar os materiais utilizados, enquanto outra observar as cores e formas. O método utilizado para definir essa perspectiva foi o de Kawakita Jiro Scupin (1997). Esse método busca organizar imagens em grupos em três fases: captura, agrupamento e categorização. Ao final é formado um diagrama de afinidades (Figura 26).

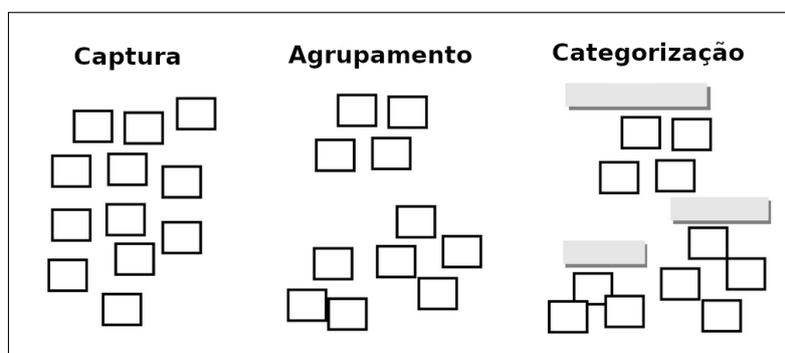


Figura 26. Método Kawakita Jiro.

Fonte: O autor.

Na fase de captura, foram produzidos recortes de imagens focando nas interfaces dos brinquedos de programar. Essas imagens foram obtidas ao parar vídeos de demonstração dos brinquedos. Após a captura, as imagens foram distribuídas em um software online e agrupadas por dois pesquisadores. Possíveis divergências, por exemplo, se blocos com texto e ícones pertencem a uma mesma categoria, foram discutidas e sanadas (Figura 27).

O processo resultou em 2 categorias, cada uma com 5 subcategorias. A primeira categoria são as interfaces virtuais, onde há presença de uma tela bidimensional onde os comandos aparecem. A segunda categoria são as interfaces tangíveis, em que a interação ocorre com objetos físicos, como blocos de encaixar ou botões acoplados no brinquedo.

As subcategorias elencadas das interfaces virtuais foram:

Código textual: Interfaces que usam código digitado em teclados, como as linguagens de propósito geral.

Blocos virtuais: Peças de quebra-cabeça que conectadas representam estruturas de programação.

Máquinas de estado: Formas geométricas conectadas por linhas, em que cada forma agrupa ações do brinquedo. As linhas representam eventos que provocam a transição entre os estados.

Linhas: Uma linha é desenhada na tela do dispositivo e o brinquedo se move conforme o seu formato.

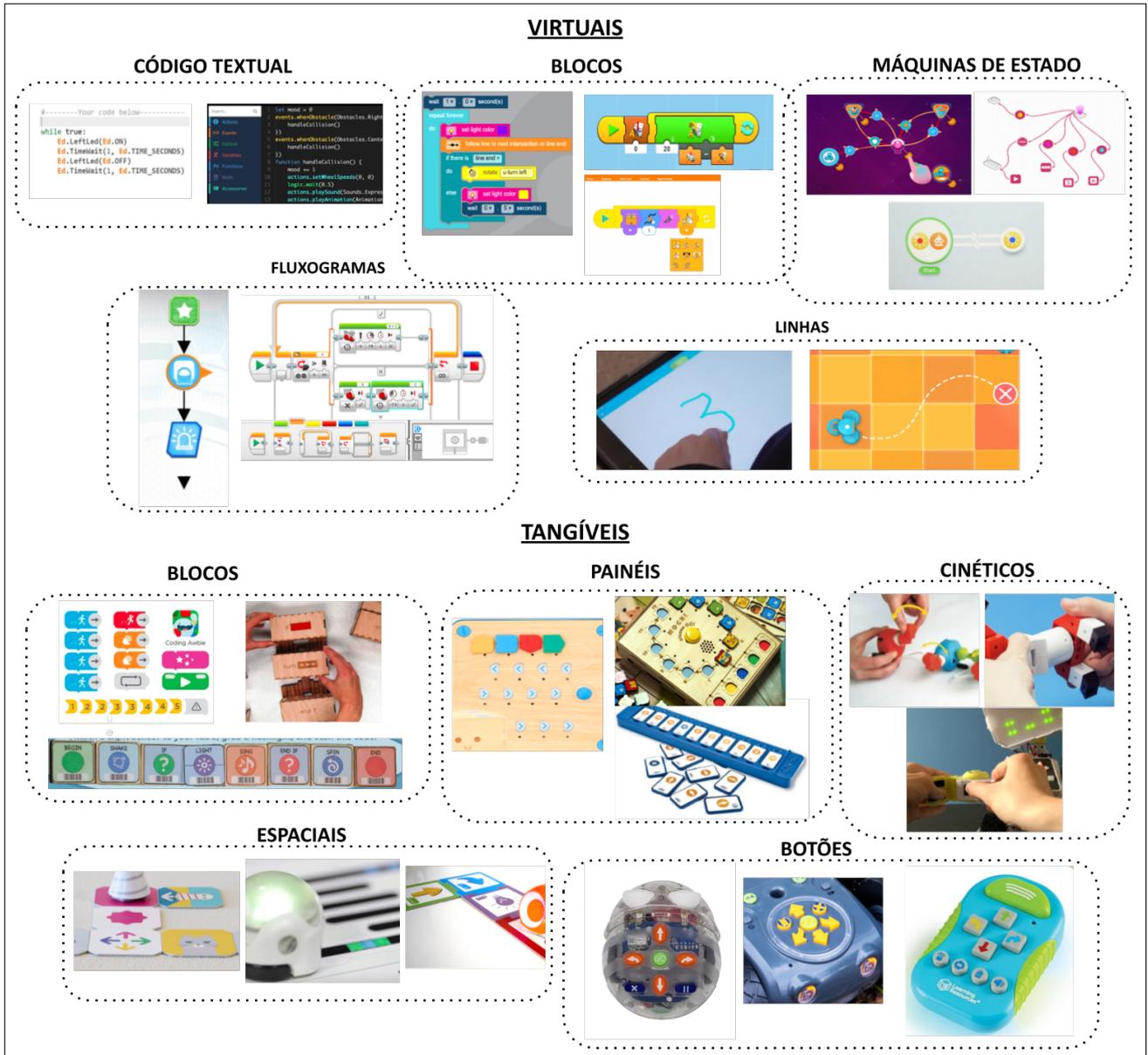


Figura 27. Agrupamento das interfaces.

Fluxogramas: Formas geométricas conectadas por linhas. Cada forma representa uma ação ou uma decisão.

As subcategorias de interface tangível são semelhantes.

Blocos físicos: Blocos físicos que podem ser interconectados e não precisam ser fixados a um painel.
Exemplos: Code a Pillar e KIBO.

Painéis: Placas de madeira ou de plástico com furos para encaixe de blocos de programação. Exemplos: Cubetto e Mojobot.

Cinéticos: Brinquedos que repetem os movimentos feitos em partes do corpo do brinquedo. Exemplo: Topobo.

Espacial: Brinquedo lê comandos do ambiente usando sensores.

Botões: Botões físicos acoplados ao corpo do brinquedo ou em um controle remoto.

Considerando essa taxonomia, o tipo mais comum de interface de brinquedos programáveis são os blocos virtuais, usados por 30 brinquedos. O segundo tipo mais usado são os botões. Outra tendência observada é o uso de mais de um tipo de interface pelo mesmo brinquedo, a fim de atender um público mais amplo quanto à faixa etária. A Figura 28 descreve o uso de cada um dos tipos de interface.

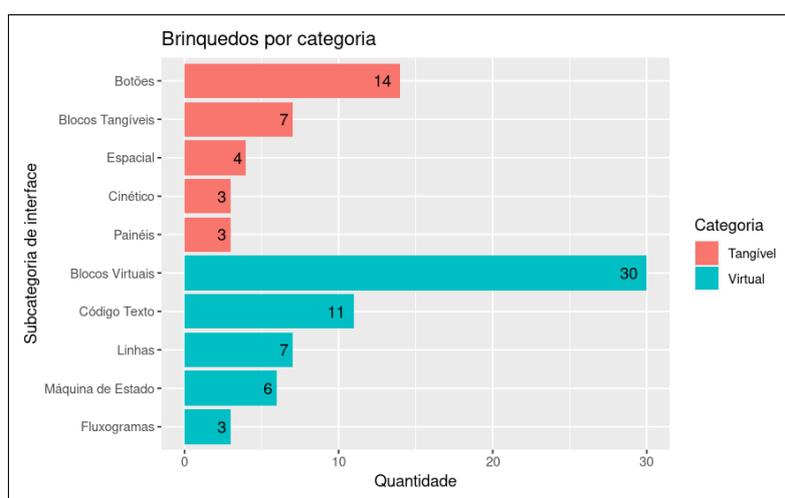


Figura 28. Brinquedos por tipo de interface.

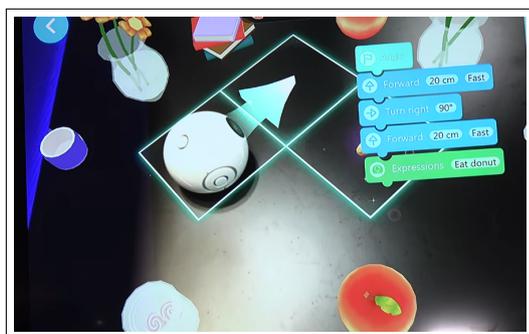
Fonte: O autor.

3.3.6 QE2 - Como a realidade aumentada é aplicada em brinquedos programáveis?

A RA, no sentido de combinar objetos virtuais e físicos, é pouco usada em brinquedos programáveis. Dos 56 brinquedos encontrados, apenas 2 seguem essa abordagem: o Botzees e o Augie. Ambos os brinquedos são da empresa *Pai Technology*, a qual apresenta o uso de RA como um diferencial da marca.

O tipo de RA utilizado é o *window-on-the-world*, ou seja, um *tablet* ou *smartphone* captura a imagem do brinquedo e associa objetos virtuais à imagem real na tela. No caso do Augie, o brinquedo é captado pela câmera e, ao ter sua posição identificada, os personagens aparecem na tela simulando uma batalha com o brinquedo físico. A RA também é usada para mostrar um mapa virtual na tela do *tablet*, e o brinquedo físico percorre esse mapa (Figura 29a). O mesmo princípio se aplica ao Botzees. O *smartphone* capta a imagem do brinquedo e mostra a imagem deste ao lado dos blocos virtuais (Figura 29b).

Uma vantagem possível dessa abordagem é a proximidade com que o brinquedo físico e os blocos virtuais aparecem na tela. Isso diminui a necessidade de olhar para dois lugares ao mesmo tempo, ou seja, olhar para a tela do aplicativo e para o ambiente real. Porém não foram encontrados estudos que evidenciem benefícios deste tipo de interface para o aprendizado de programação por crianças.



(a) Programação do Augie com RA



(b) Programação do Botzees com RA

Figura 29. Usos de RA com brinquedos programáveis.

3.3.7 QE3 - Quais são os conceitos de algoritmos abordados nas interfaces de brinquedos programáveis?

A abrangência de conceitos de algoritmos abordados por brinquedos programáveis foi analisada nos seguintes conceitos: variáveis, estruturas de repetição, condicionais, listas, recursividade, par-

alelismo, sequenciamento, funções, parâmetros e eventos. Aspectos como tipos de dados, constantes, e paradigmas de programação não foram abordados dado que são conceitos abstraídos nessas interfaces.

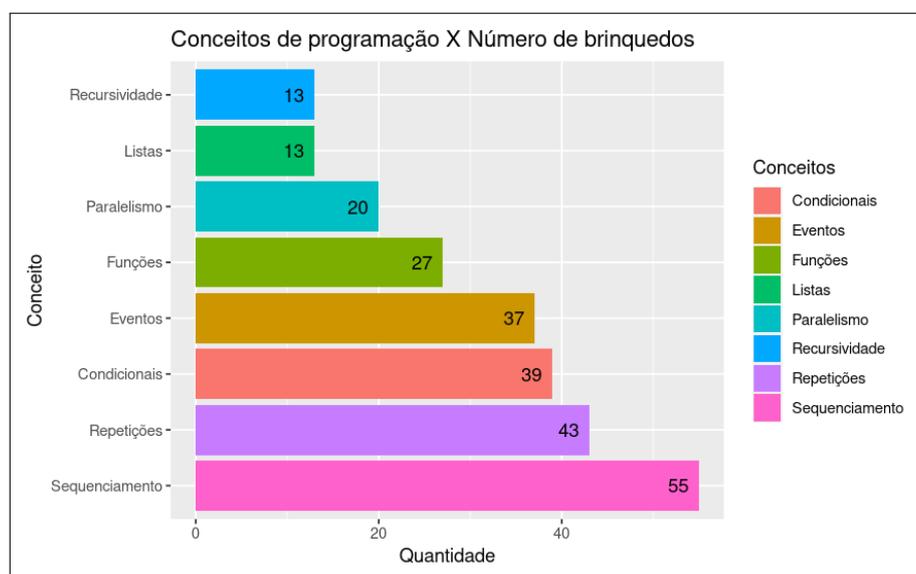


Figura 30. Conceitos de programação abordados por brinquedos programáveis.

Fonte: O autor.

A Figura 30 apresenta o resultado da análise. Como o esperado, o sequenciamento é o conceito mais abordado: criar sequências de comandos a serem executados é intrínseco dos brinquedos programáveis. O segundo conceito mais abordado (76% dos brinquedos) são repetições. As estruturas de repetição são normalmente suportadas em interfaces virtuais de programação em blocos.

Outros dois conceitos interligados são condicionais e eventos. 66% dos brinquedos disparam algum evento captado por sensores de luminosidade, proximidade, sons, etc. Esses eventos são usados em condicionais para ações do tipo "se obstáculo à frente, vire à esquerda", "se luminosidade maior que 30, toque som", entre outros.

Conceitos mais avançados, como paralelismo, recursividade e vetores são suportados pelos brinquedos com interfaces de código textual, com suporte a linguagens de propósito geral. Recursividade e vetores são conceitos suportados por 23% dos brinquedos.

3.4 CONSIDERAÇÕES

Por fim, esse capítulo ressalta três aspectos. (i) O mapeamento industrial não apontou uso de realidade aumentada projetiva com brinquedos programáveis, ao menos em nível comercial. Entretanto (ii) os trabalhos relacionados demonstram que usar projeção em interfaces de programação é viável e

abre novas possibilidades de interação, como a programação espacial(BURLESON et al., 2018). Além disso, o mapeamento indicou (iii) uso de interfaces tangíveis por 55% dos brinquedos. A proposta apresentada no Capítulo 4 considera esses três aspectos e busca representar um tipo inovador de interação, ou, ao menos, contribuir com pesquisas que indiquem a adequação da realidade aumentada projetiva aplicada às interfaces de brinquedos programáveis.

4 DESENVOLVIMENTO

Este capítulo apresenta a interface desenvolvida, denominada RoPE AR (Figura 31). Primeiramente a Seção 4.1 descreve a interface do ponto de vista do usuário, ou seja, os elementos com os quais a criança ou o adulto interagem. A seguir a Seção 4.2 traz detalhes técnicos das tecnologias utilizadas, como a comunicação Bluetooth entre smartphone e robô.



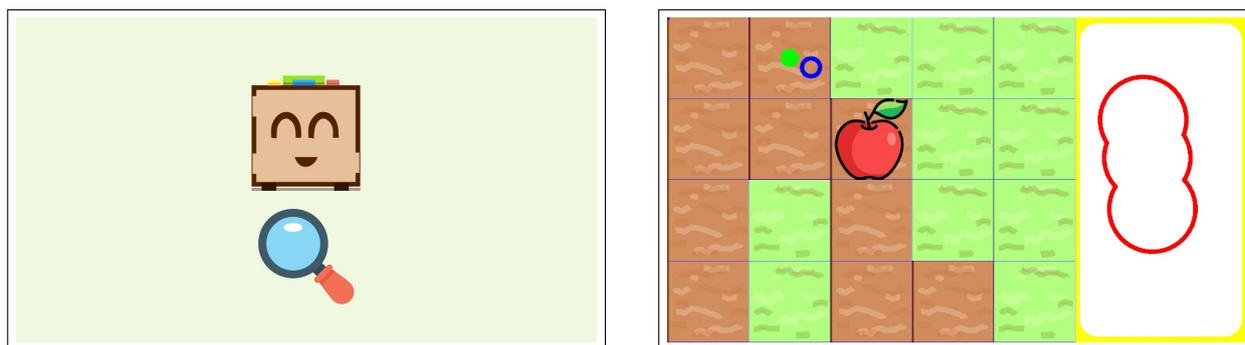
Figura 31. RoPE AR em uso. Círculo projetado mostra bloco que forma o algoritmo.
Fonte: O autor.

4.1 ELEMENTOS DA INTERFACE

4.1.1 Aplicativo

A interação inicial ocorre quando um adulto usa o aplicativo para conectar o smartphone com o RoPE. Ele é constituído de apenas duas telas. A primeira tela permite conectar o smartphone ao brinquedo (Figura 32a). Ela mostra um desenho do RoPE e uma lupa indicando a procura pelo brinquedo, ou seja, a conexão com o brinquedo físico. Gravações de uma voz guiam o usuário a ligar o RoPE acionando seu botão inferior e informam quando a conexão foi estabelecida. Ao se conectar o brinquedo também emite um *feedback* sonoro e luminoso.

A segunda tela aparece após a conexão com o brinquedo. Ela apresenta um desafio contendo uma sequência de mapas (Figura 32b). Cada mapa tem um caminho, uma posição inicial (círculo verde) e um ou mais itens coletáveis. Na direita da tela há um retângulo branco. Ao ser projetada, essa cor não interfere nas cores dos blocos físicos. Manter as cores originais dos blocos é importante para não interferir na identificação das marcas fiduciais que são preto e branco. Por fim, o último elemento



(a) Tela de conexão.

(b) Tela de desafio.

Figura 32. Telas do aplicativo.

Fonte: O autor.

da tela são círculos vermelhos que indicam quais blocos fazem parte do algoritmo a ser executado pelo brinquedo.

Relembrando, a criança não interage com o aplicativo. Toda interação ocorre com o RoPE e com blocos físicos, posicionados no chão. A criança cria algoritmos ao sequenciar os blocos físicos, identificados pelo App. Caso exista um algoritmo válido, a luz emitida pelo projetor destaca os blocos do algoritmo.

O brinquedo executa o algoritmo quando a criança pressiona seu botão verde. Ao executar o programa, o brinquedo se move e o aplicativo capta as posições percorridas. Deste modo, consegue identificar se o brinquedo colidiu ou não nos itens coletáveis do mapa. Quando o brinquedo coleta todos os itens, o aplicativo reproduz uma gravação de parabenização, e mostra o mapa seguinte. Caso não existam outros mapas, uma gravação sinaliza que a brincadeira acabou.

4.1.2 Elementos tangíveis

A interface possui quatro elementos tangíveis: o brinquedo RoPE, os blocos de código, os marcadores de calibragem, e o ativador de depuração (Figura 33).

1. **Brinquedo RoPE:** Ao brinquedo foi adicionado um identificador fiducial que auxilia a determinar sua posição. A informação da posição do brinquedo permite à interface reagir a seus movimentos, disparando eventos sonoros e animações. Além da marca fiducial, o brinquedo tem os botões superiores. Por meio do botão verde do brinquedo a criança inicia a execução do algoritmo programado com os blocos de papelão. Os demais botões são usados no modo de depuração. Neste modo, o brinquedo para antes de cada movimento e acende a seta indicando

qual será a próxima ação. O brinquedo continua a execução quando o botão é clicado, e assim a criança pode observar a execução passo a passo.

2. **Blocos de código:** Há dois tipos de blocos: os direcionais e o bloco de início. Os blocos direcionais tem desenhos que mostram o RoPE fazendo o movimento de andar para frente, andar para trás, virar a esquerda, e virar à direita. O bloco de início corresponde ao botão verde do RoPE e indica o início da execução. Além disso, o bloco de início marca a posição onde os blocos direcionais devem ser encaixados para formar o algoritmo. O formato curvado busca sugerir como deve ser o encaixe entre os blocos. Diferente dos blocos direcionais, o bloco de início tem formato diferenciado para impedir o encaixe de mais de um bloco.
3. **Marcadores de calibragem:** Marcas fiduciais que identificam os quatro cantos da área projetada. Para projetar os elementos na posição correta o algoritmo precisa mapear a posição de projeção, bem como identificar a distorção de perspectiva resultante. Um adulto responsável por preparar a interface coloca os marcadores nos cantos da área projetada, e a câmera do *smartphone* capta a posição da mesma. Após a captação os marcadores são retirados.
4. **Ativador de depuração:** O quarto elemento tangível é representado por uma joaninha (em inglês *ladybug*). Essa figura foi selecionada por ser uma figura lúdica e também por ser um inseto, uma referência à origem do termo *bug*. Quando a joaninha está visível o brinquedo RoPE entra em modo de depuração, onde a execução de cada comando exige que a criança aperte o botão correspondente. Quando não está visível, o brinquedo executa a sequência completa de comandos sem interrupção.

Todos os elementos tangíveis possuem marcas fiduciais de 4 cm de diâmetro. Esse tamanho facilita a identificação por uma câmera que gera imagens de 1280x720 pixels durante gravação de vídeo. A medida de 4 cm foi obtida empiricamente, iniciando com 3 cm de diâmetro e aumentando até a identificação de todas as marcas presentes no campo de visão da câmera. Um último detalhe é as marcas serem impressas em papel não reflexivo. Materiais reflexivos não interferem na captação das cores na superfície dos blocos pela câmera, e por consequência não prejudicam a identificação das marcas fiduciais presentes nessa superfície.

4.2 ASPECTOS TÉCNICOS

A apresentação dos aspectos técnicos será feita com base na arquitetura de software do projeto. Essa arquitetura possui quatro dispositivos: um *smartphone*, o brinquedo RoPE, um servidor web



Figura 33. Elementos tangíveis. 1 - RoPE. 2 - Blocos de código. 3 - Marcadores de calibragem. 4 - Ativador de depuração.

e um projetor (Figura 34). A Subseção 4.2.1 apresenta o smartphone e os módulos do aplicativo implementado. A Subseção 4.2.2 apresenta a comunicação com o brinquedo RoPE. O Apêndice E apresenta detalhes do mini projetor utilizado. O servidor web não será abordado por que está fora do escopo deste trabalho.

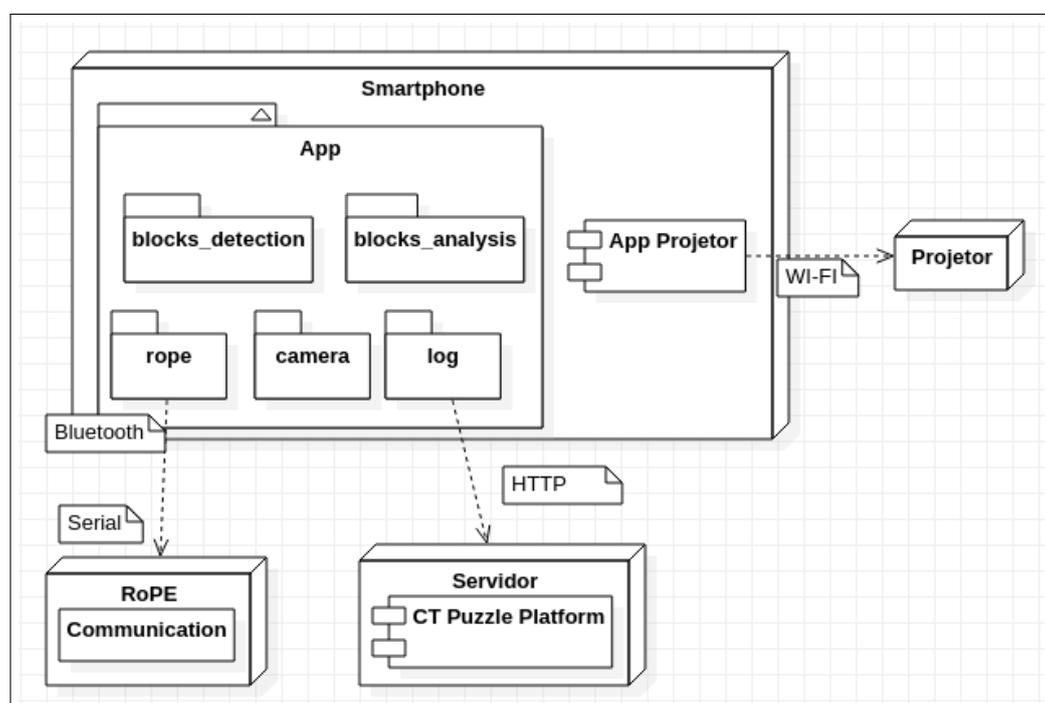


Figura 34. Visão geral dos componentes do sistema.

4.2.1 Smartphone

Um smartphone com sistema operacional Android é o componente central, pois coordena a comunicação com o projetor e o servidor web. Dois aplicativos permanecem ativos no smartphone. O primeiro aplicativo é o Mirroring 360, que transmite a tela do smartphone ao projetor via Wi-Fi. Esse aplicativo está disponível na loja de aplicativos da Google, e há diversos aplicativos similares que executam a mesma função. O Mirroring 360 foi selecionado por ser gratuito e não mostrar anúncios.

O segundo aplicativo foi desenvolvido neste trabalho. Ele tem as funções de:

- gerar o cenário virtual
- captar os blocos tangíveis e, se houver um algoritmo, enviá-lo ao RoPE
- detectar colisões entre o RoPE e os elementos virtuais coletáveis
- enviar os eventos ocorridos para a CtPuzzle Platform

Para isso, o aplicativo é dividido em cinco módulos: Câmera, Detecção de blocos, Análise dos blocos, Registro de interações, e Comunicação com o RoPE . As próximas quatro seções apresentam detalhes de implementação dos quatro primeiros módulos. O quinto módulo (comunicação) é abordado na Subseção 4.2.2.

4.2.1.1 Módulo de Câmera

A câmera capta as imagens utilizadas na identificação dos elementos tangíveis, com os quais a criança interage. Para essa captação o sistema operacional Android oferece dois modos de acesso à câmera: fotografia e vídeo. O modo fotografia obtém imagens com mais resolução, porém imagens maiores também precisam de mais tempo de processamento ¹. O modo vídeo obtém um fluxo contínuo de imagens, porém em menor resolução. Esta pesquisa usou dois smartphones: um Meizu M5c e um Xiaomi Redmi 8. A Figura 35 mostra, em proporção, as resoluções disponíveis para os dois smartphones utilizados.

No modo vídeo o M5c produz imagens de 1280 x 720 pixels, e o Redmi 8 produz imagens de 1920 x 1080 pixels. Já no modo foto, o M5c produz imagens de 3264 x 2448 pixels, e o Redmi 8

¹ O tempo entre captura e processamento ficou entre 0.5 a 1 segundo no modo fotografia. Essa medição ocorreu por meio de logs com o dispositivo M5c.

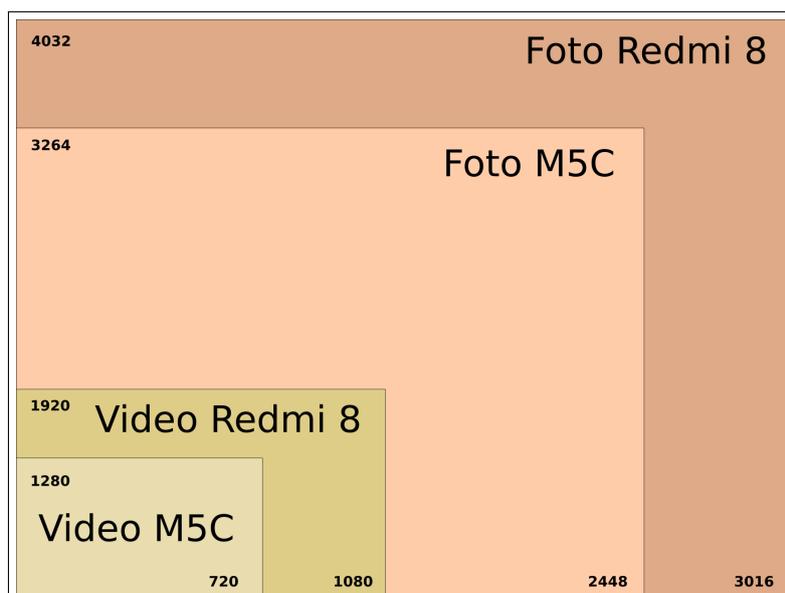


Figura 35. Comparativo de dimensões de foto e vídeo para os smartphones utilizados.

produz imagens de 4032 x 3016 pixels. Após testes empíricos nos dois modos disponíveis, o modo vídeo no Redmi 8 apresentou o maior desempenho entre assertividade na identificação dos blocos e tempo de processamento.

O modo vídeo, entretanto, produz imagens em formato incompatível com exigido pela biblioteca TopCodes. Enquanto a TopCodes processa imagens em formato RGB², o formato gerado por vídeo é o YUV³. Portanto, é preciso converter o formato YUV para RGB, o que ocorre com a seguinte fórmula:

$$\begin{aligned}
 R' &= Y + 1.140V \\
 G' &= Y - 0.395U - 0.581V \\
 B' &= Y + 2.032U
 \end{aligned}
 \tag{4.1}$$

A Equação 4.1 precisa ser invocada para cada pixel. Para uma imagem de 1920 x 1080 isso significa iterar em um laço de repetição 2 073 600 vezes. Entretanto, esse processo não ocorre apenas

² O RGB (*Red, Green, Blue* - Vermelho, Verde e Azul) armazena as informações de cada pixel em 3 bytes, um para R, outro para G e outro para B. O valor de cada byte varia entre 0 e 255, onde 0 significa ausência de cor e 255 significa presença de cor. A cor preta é representada pela ausência das três cores, igual a 0 0 0, e cor branca é a presença total das três cores, igual a 255 255 255. Nesta estratégia cada um dos três bytes tem a informação de luminosidade.

³ Y significa luminosidade, e U e V representam intensidade das cores. Essa separação entre luminosidade e cores ocorreu por razões históricas. Os televisores em preto e branco necessitavam apenas da informação de luminosidade, e os televisores coloridos, que surgiram posteriormente, precisavam de canais de cores (JACK, 2001). O formato YUV manteve o funcionamento das televisões em preto e branco, que decodificavam apenas o canal Y, enquanto os aparelhos coloridos usaram também as informações de cores.

em uma imagem, mas sim em um fluxo contínuo de imagens.

Para que este processamento intenso não cause atrasos perceptíveis para o usuário, o módulo de câmera utiliza a API Renderscript. Essa API do sistema Android viabiliza operações computacionais de alto desempenho, como é o processamento de imagens. O ganho de performance decorre da execução de código nativo em C++, que pode ser executado paralelamente em GPU (SAMS, 2011). Isso viabiliza a conversão de mais de um pixel simultaneamente.

4.2.1.2 Módulo de Detecção de Elementos Tangíveis

Após a captação da imagem e sua conversão para o formato RGB, ocorre a detecção das marcas fiduciais na imagem para identificar as posições dos elementos tangíveis da cena. A biblioteca TopCodes é responsável por identificar as marcas fiduciais nos objetos. Esta biblioteca possui implementações em três linguagens: Java, JavaScript e C++. O uso do algoritmo em Java se mostrou inviável devido à lentidão de cerca de 3 segundos para detectar os topcodes em uma imagem de 1280 x 720. Com isso restou o uso do algoritmo em C++.

A tecnologia que viabiliza o uso de C++ em ambiente Android é *Java Native Interface* (JNI). Para utilizar código nativo, uma função do tipo `external` é declarada no código Java ou Kotlin e sua implementação ocorre em C++. O Quadro 8 demonstra o relacionamento entre ambas as linguagens. O código Kotlin invoca a função `external`, cuja implementação é em C++. Os parâmetros que transitam entre os dois ambientes são a altura e a largura da imagem, bem como uma lista de números inteiros que representam os pixels da imagem no formato RGB.

Quadro 8. Ligação entre código nativo escrito e código Kotlin.

```
// kotlin (arquivo TopcodesScanner.kt)
private external fun searchTopCodesNative(imageWidth: Int, imageHeight:
↳ Int, imageData: IntArray) : Array<TopCode>
```

```
// c++ (arquivo native-lib.cpp)
extern "C"
JNIEXPORT jobjectArray JNICALL
Java_topcodes_TopCodesScanner_searchTopCodesNative(JNIEnv *env, __unused
jobject _, jint image_width, jint image_height, jintArray image_data)
{
    // uso da biblioteca topcodes em código nativo.
}
```

A execução do código nativo por meio da JNI possibilitou a redução do tempo de captura de

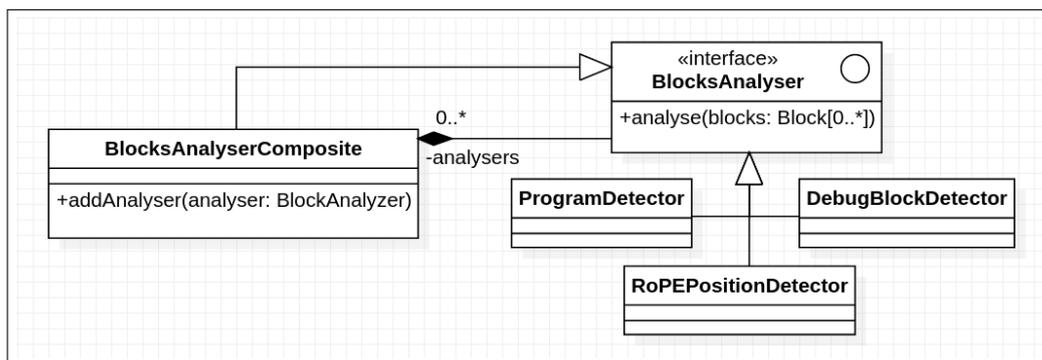


Figura 36. Aplicação do padrão Composite.

Fonte: O autor.

imagem e identificação de TopCodes para 1 019,9 milissegundos⁴, ou cerca de 1 quadro por segundo. Sem o uso de JNI a identificação custava mais de 3 segundos. Por fim, após a execução, o código nativo retorna a lista de marcas detectadas. Cada marca tem como atributo as coordenadas, ângulo, tamanho e código identificador. Essas informações podem, então, ser analisadas para compreender o posicionamento dos elementos tangíveis no cenário.

4.2.1.3 Módulo de Análise de Elementos Tangíveis

Após a identificação dos elementos tangíveis, o módulo de análise:

- analisa se um algoritmo foi construído
- detecta o posicionamento do brinquedo RoPE sobre o tapete
- detecta se o bloco ativador de depuração está visível

Estas ações são uma parte do código-fonte propensa a mudanças. Outros tangíveis podem ser criados em funcionalidades futuras, exigindo análises específicas. O padrão de projeto Composite foi a solução utilizada para facilitar a adição de novos objetos tangíveis sem alterar o código já existente. Esse padrão facilita a construção de hierarquias, construídas com grupos de objetos (composições) e objetos individuais (folhas). Tanto folhas como composições obedecem uma mesma interface, e os objetos do tipo composição podem ter listas de folhas ou outras composições (GAMMA et al., 1994).

Seguindo este padrão, o código organiza uma estrutura hierárquica de *analísadores de blocos*⁵ (ver Figura 36). A classe `BlocksAnalyserComposite` é uma composição, e as classes `ProgramDetector`,

⁴ Tempo médio para captação de uma imagem. O cálculo usou 210 capturas no smartphone Redmi 8.

⁵ Os blocos são os elementos tangíveis.

RoPEPositionDetector e DebugBlockDetector são folhas. Essas quatro classes implementam a interface BlocksAnalyser, ou seja, são analisadores de blocos. A primeira classe é um analisador composto pelos três analisadores folha. O código cliente⁶ tem apenas um objeto do tipo BlocksAnalyserComposite, o qual tem uma lista de analisadores. Quando o código cliente solicita a esse objeto que analise os blocos, este repassa a solicitação a cada um dos analisadores da lista para executarem especializadas (Quadro 9). O código cliente não sabe quais análises ocorrem, e o comportamento do programa pode ser alterado ao adicionar mais analisadores na composição. Isso evita alterações em cada um dos analisadores, que ficam responsáveis por ações específicas, e também evita alterações no código cliente, pois a interface com o composite se mantém inalterada.

Quadro 9. Implementação da classe de composição.

```
class BlocksAnalyzerComposite : BlocksAnalyzer {
    private val analyzers = mutableListOf<BlocksAnalyzer>()

    override fun analyze(blocks: List<Block>) {
        if(blocks.isNotEmpty()){
            // A composição repassa a solicitação de análise a cada um
            ↪ dos analisadores da sua lista.
            analyzers.forEach {
                it.analyze(blocks)
            }
        }
    }

    fun addBlocksAnalyzer(analyser: BlocksAnalyzer) =
        ↪ analyzers.add(analyser)
}
```

4.2.1.4 Módulo de Registro de Interações - CtPuzzle Platform

Este módulo interage com a plataforma web CtPuzzle Platform⁷. Esta plataforma permite a criação de testes de avaliação do Pensamento Computacional através da definição de **mecânicas** de puzzles⁸. Uma mecânica tem conjunto de atributos que especificam de maneira abstrata as regras do puzzle. No caso desta pesquisa, o puzzle consiste em um mapa, um caminho a ser percorrido, posições de objetos a serem coletados, e um conjunto de comandos esperados como solução.

Após a definição dos atributos do puzzle, a plataforma permite definir instâncias desse puzzle,

⁶ Código que solicita a análise dos blocos.

⁷ <ctplatform.playerweb.com.br>

⁸ Desafio ou jogo de lógica.

denominadas **itens**. Como fases de um jogo, cada item tem valores concretos conforme as definições da mecânica⁹. Como descrito, neste trabalho, a mecânica define que o puzzle deve ter objetos coletáveis posicionados em algum ponto do mapa. Um item especifica que haverá um objeto coletável na posição (2,3) do mapa, e que este mapa terá uma dimensão de 4 por 5, por exemplo.

Tendo por base as regras definidas nas mecânicas, e os valores dos atributos para cada item, o aplicativo busca essas informações na plataforma e gera a interface do jogo. Na plataforma estão definidos 9 itens/fases, organizadas em ordem crescente de dificuldade. O aplicativo interpreta essa configuração e mostra as fases para o usuário. Por fim, o aplicativo envia à plataforma as respostas para cada um dos itens testados. Cada resposta possui uma lista de tentativas de solução do desafio, ou seja, os algoritmos executados pelo robô. Isso permite entender como as estratégias de resolução mudaram até a obtenção da solução final.

4.2.2 RoPE: Comunicação Bluetooth

Para entender como ocorre a comunicação com o brinquedo RoPE via Bluetooth é necessário entender a organização do seu *firmware* e o conceito de comunicação serial. O *firmware* é o programa instalado no chip ATmega328p, o qual é acoplado à placa principal e coordena as reações aos eventos percebidos pelo brinquedo. As reações possíveis são o acionamento de luzes, o acionamento dos motores e a emissão de sons. O *firmware* coordena o acionamento dessas ações em eventos de cliques nos botões e de recepção de mensagens na comunicação serial do Arduino.

A comunicação serial possibilita a transferência de dados entre uma placa Arduino e outros computadores ou dispositivos. Todas as placas Arduino possuem ao menos dois pinos para essa comunicação, um para leitura e outro para a escrita. Normalmente esses pinos são usados durante a atualização do *firmware*, quando o programa construído em ambiente de desenvolvimento é instalado no chip.

Além disso, esses mesmos pinos permitem a comunicação sem fio por meio da adaptação de um módulo, que pode ser Bluetooth, Wi-Fi, entre outros. Os módulos são dispositivos de hardware que possibilitam a adição de novas funcionalidades a placas Arduino. Um destes módulos é HM-10, destinado a habilitar comunicação Bluetooth de baixo consumo de energia (BLE - *Bluetooth Low Energy*). Esse módulo possui os pinos de energia (VCC), aterramento (GND), leitura (RX) e escrita

⁹ Pode-se comparar mecânica e item com classe e objeto, na programação orientada a objetos. A classe/mecânica define atributos, e o objeto/item tem valores concretos para estes atributos.

(TX). Os pinos de leitura e escrita são conectados aos pinos de escrita e leitura do brinquedo RoPE, o que habilita a transferência de dados recebidos via Bluetooth pelo módulo, o qual os repassa via serial para o chip do brinquedo (Figura 37).

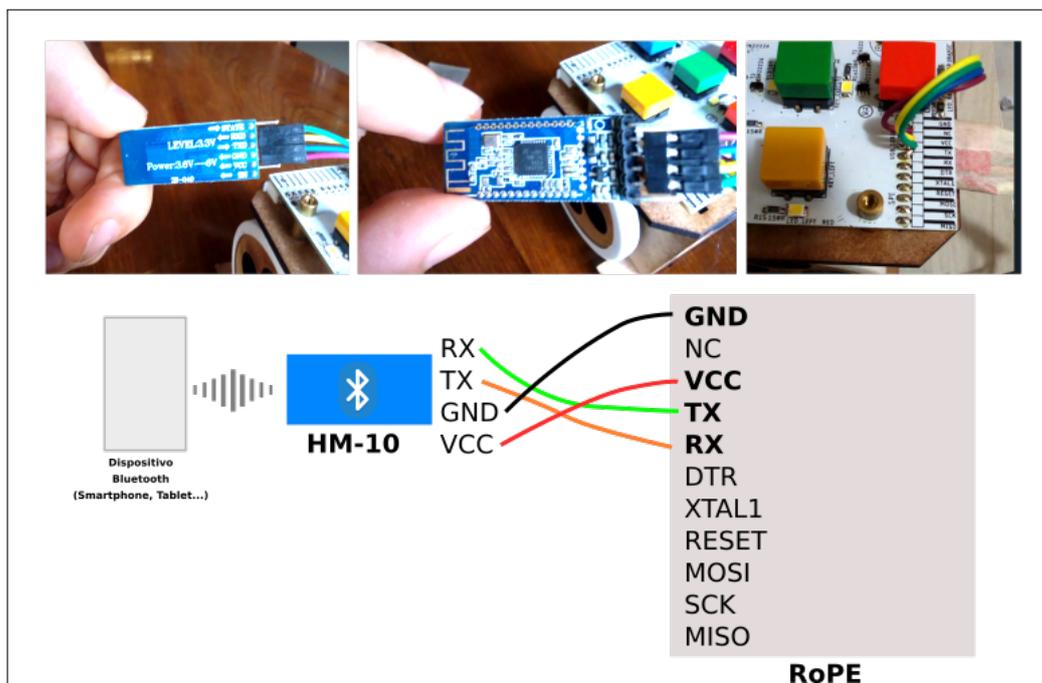


Figura 37. Conexão entre o módulo HM-10 e a placa do brinquedo RoPE.

Fonte: O autor.

O *firmware* é responsável por interpretar os dados recebidos na serial. Este é implementado seguindo o paradigma Orientado a Objetos, de modo a mapear as responsabilidades e componentes físicos existentes no brinquedo. Neste sentido, o código-fonte do *firmware* tem classes que correspondem aos componentes de *hardware* como buzzer, CPU, teclado e bateria (Figura 38). A classe *Battery*, por exemplo, monitora o nível da bateria e impede o acionamento do brinquedo quando há pouca energia armazenada.

Seguindo este paradigma, este trabalho adicionou ao código existente uma classe responsável pela comunicação, ou seja, recepção, interpretação e envio de dados por meio da comunicação serial. Essa classe de comunicação permite o registro de caracteres que, ao serem captados, acionam funcionalidades do brinquedo. A recepção da letra *f*, por exemplo, dispara a ação *adicionar o comando "frente" na sequência de ações*.

Um protocolo de comunicação define os caracteres a serem enviados, bem como uma sequência de caracteres para definir o início de uma mensagem (Quadro 10). Além disso, o protocolo define mensagens que informam eventos ocorridos no brinquedo. Isto possibilita que aplicações externas

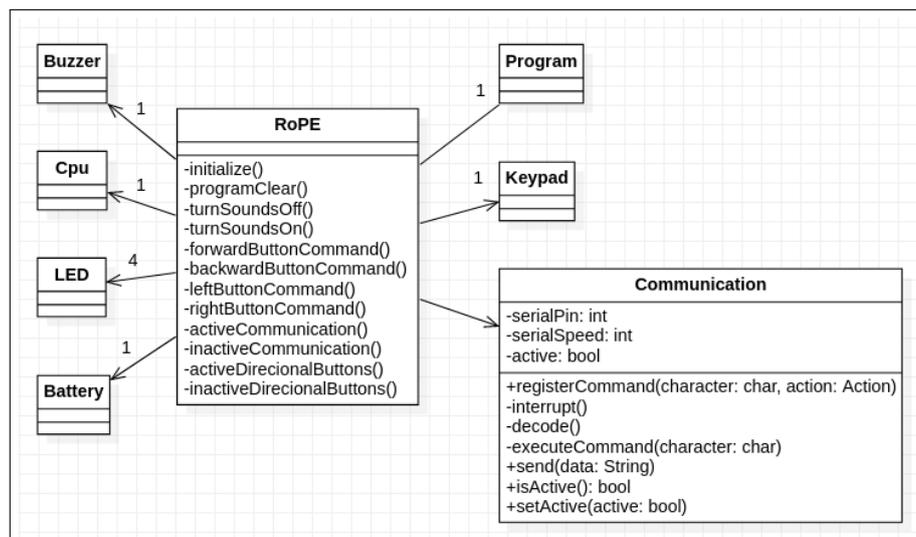


Figura 38. Diagrama de classes do firmware.

Fonte: O autor.

recebam estes eventos e reflitam mudanças de estado do brinquedo. Com isso um aplicativo de programação em blocos pode destacar um bloco em execução, por exemplo.

Ao registrar funções disparadas ao receber caracteres, o código usa o padrão *Observer* (GAMMA et al., 1994), o que facilita registrar novos caracteres para alterar o comportamento do robô. A limitação do registro de apenas um caractere é que uma letra apenas identifica a função a ser invocada, mas não informa seus parâmetros. Isso permite, por exemplo, acionar a função de giro, mas não definir o seu ângulo. Para isso seria necessário o envio de uma letra para identificar a função a ser invocada, seguida de uma lista de parâmetros para a função. A passagem de parâmetros não foi implementada, dado que valores constantes, como giro de 90 graus e avanço de 12 cm atendem as necessidades deste trabalho.

4.3 CONSIDERAÇÕES

Este capítulo descreveu os elementos da interface RoPE AR. Iniciou apresentando o aplicativo para Android e elementos tangíveis, com os quais o usuário interage. A seguir apresentou aspectos técnicos da implementação. Entre os aspectos está a organização em módulos, responsáveis por capturar a imagem, processá-la, e gerar o cenário conforme os itens definidos na CtPuzzle Platform, e também registrar as respostas na plataforma. Outro aspecto é o uso de tecnologias específicas do ambiente Android, como RenderScript e JNI. Essas tecnologias viabilizaram a captura, conversão de formato de imagem e identificação das marcas fiduciais. O último aspecto apresentado foi a comunicação com

Quadro 10. Protocolo de comunicação

Mensagens externas para o RoPE	
String	Ação do RoPE
cmds:	Indica início de mensagem
c	Apaga o programa da memória
s	Desativa sons
S	Ativa sons
f	Adiciona comando "frente" programa
b	Adiciona comando "trás" programa
r	Adiciona comando "direita" no programa
l	Adiciona comando "esquerda" programa
e	Executa programa
a	Desativa conexão Bluetooth
x	Desativa botões direcionais
X	Ativa botões direcionais
Mensagens do RoPE para outros dispositivos	
Expressão regular	Mudança de estado do brinquedo
<addi:(f b l l)>	Comando "frente", "trás", "direita" ou "esquerda" adicionado
<start_required>	Botão "iniciar" pressionado e o brinquedo está em estado de programação externa
<program:(started terminated)>	Notifica início e término de execução.
<executed:[0-9]1,2>	Notifica início da execução de um comando.

Fonte: O autor.

Quadro 11. Ligação entre caracteres e funções executadas no brinquedo.

```

communication.registerCommand('c', programClear);
communication.registerCommand('s', turnSoundsOff);
communication.registerCommand('S', turnSoundsOn);
communication.registerCommand('f', forwardButtonCommand);
communication.registerCommand('l', leftButtonCommand);
communication.registerCommand('r', rightButtonCommand);
communication.registerCommand('b', backwardButtonCommand);
communication.registerCommand('a', inactiveCommunication);
communication.registerCommand('A', activeCommunication);
communication.registerCommand('x', inactiveDirectionalButtons);
communication.registerCommand('X', activeDirectionalButtons);
communication.registerCommand('e', executeButtonExternalAction);

```

o RoPE via Bluetooth. Esta foi viabilizada pela adaptação do módulo HM-10 e a definição de um protocolo de comunicação.

Outras tecnologias e dispositivos poderiam ter sido utilizados como alternativa aos aqui propostos. Em vez do smartphone, poderia ser usada uma câmera e um notebook. No lugar da TopCodes,

poderia ser usado um algoritmo de identificação de objetos independente de marcas fiduciais. Independente de dispositivo ou algoritmo usado, porém, os requisitos de captura de imagem, identificação dos objetos tangíveis e comunicação como o RoPE se manteriam. A escolha das tecnologias utilizadas se deu por serem acessíveis, abertas, gratuitas, e capazes de atender aos requisitos do trabalho.

O próximo capítulo (Capítulo 5) apresenta o contexto e as diretrizes para avaliação da RoPE AR. A avaliação é qualitativa exploratória, e observa o uso da RoPE AR por crianças e capta as opiniões de professoras. A avaliação qualitativa exploratória se justifica pela inovação da proposta. Apesar de os trabalhos similares unirem tecnologias de projeção e tangíveis, a observação de crianças construindo algoritmos com blocos associados com RA é ainda um fenômeno a ser explorado. Os resultados obtidos com a avaliação são descritos no Capítulo 6.

5 AVALIAÇÃO

Essa pesquisa seguiu uma abordagem qualitativa exploratória. O objetivo foi descrever as interações das crianças de 4 a 6 anos com uma interface de RA projetiva durante a programação e depuração de algoritmos em um ambiente educacional infantil. Além disso, buscou captar as percepções de professores a respeito da proposta. Neste sentido, a Seção 5.1 descreve o contexto da coleta de dados, a Seção 5.2 descreve os participantes, e a Seção 5.3 detalha o ambiente escolar. Por fim, a Seção 5.4 apresenta os instrumentos de coleta.

5.1 CONTEXTO

5.1.1 Município

A coleta de dados ocorreu em um Centro de Desenvolvimento Infantil (CDI) no município de Gaspar, no estado de Santa Catarina. Dados do Instituto Brasileiro de Geografia e Estatística (IBGE) permitem comparar o município em relação aos demais municípios do Brasil¹. A cidade está entre as 10% mais populosas do Brasil (70 mil habitantes), e entre as cidades brasileiras com maior nível de ocupação (41,8% da população).

Gaspar tem uma taxa de 9,05 mortes a cada 1000 nascidos vivos, o que a coloca na metade inferior do ranking de municípios do Brasil. Por outro lado, está entre os 10% de cidades brasileiras com maior cobertura de esgotamento sanitário (87,3%).

Em aspectos educacionais, a taxa de escolarização entre 6 e 14 anos é 93,7%, próximo da média brasileira. O desempenho no Índice de Desenvolvimento da Educação Básica (IDEB) para anos iniciais é maior que 74% dos municípios brasileiros, tendo nota 6,3. Essa nota diminui para 4,7 nos anos finais, tendência comum no Brasil.

Com isso, em relação às demais cidades brasileiras, percebe-se que Gaspar é uma cidade populosa, com condições sanitárias adequadas, escolarização média e desenvolvimento educacional de médio a alto. Certamente esses dados não permitem uma visão completa ou profunda do contexto municipal, mas permitem posicioná-lo em aspectos de população, saúde e educação em relação ao Brasil. A próxima seção aprofunda a descrição do CDI onde a pesquisa ocorreu.

¹ <<https://cidades.ibge.gov.br/brasil/sc/gaspar/panorama>>

5.1.2 Centro de Desenvolvimento Infantil

A pesquisa ocorreu em um CDI de um bairro urbano, distante cinco quilômetros do centro da cidade. O local foi selecionado por conveniência, devido ao contato do pesquisador com uma professora que leciona no local. A pesquisa, entretanto, não ocorreu nas turmas desta professora, e nenhum dos participantes conhecia o brinquedo RoPE. Outro critério para a seleção do centro foi ter internet disponível, dado que o projetor e a Ct Puzzle Platform são acessíveis por conexão sem fio local.

O Projeto Político Pedagógico (PPP) do CDI apresenta dados históricos e da estrutura física do local. O mesmo foi fundado em 1990, sendo uma parceria entre a prefeitura municipal e a associação de moradores do bairro. Atualmente possui 38 colaboradores, que promovem o atendimento de 246 crianças. A idade atendida vai de zero e cinco anos e onze meses, e há 12 turmas organizadas em três grupos etários: Infância I (0 a 2 anos), Infância II (2 a 4 anos) e Infância III (4 a 6 anos). Dentro de cada grupo etário as turmas também são divididas. Na Infância III, por exemplo, há turmas de 4 a 5 e de 5 a 6 anos. O espaço tem 9 salas onde as professoras realizam suas atividades, uma biblioteca com cerca de 300 livros infantis, um parquinho, banheiros, sala de coordenação e sala do zelador. Além das salas existentes, está em curso a construção de novos espaços no local onde antes havia o parquinho, o qual foi deslocado para um espaço ao lado do CDI.

O PPP do CDI apresenta um diagnóstico da comunidade, construído a partir de um questionário. O trabalho, com data de 2020, apresenta gráficos analisados aqui visualmente, pois não apresentam seus números (Figura 39). Dois terços das crianças moram com pai e mãe, metade vai de carro ao CDI, e quase a metade não tem computador em casa. A respeito dos pais, a maior parte auxilia os filhos nas atividades do CDI mas não tem disponibilidade em horário comercial. A renda de metade das famílias é de 1 a 2 salários-mínimos, e a outra metade vai de 2 a 4.

Considerando os dados de ocupação da população, providos pelo IBGE, e os gráficos apresentados no PPP, infere-se que as condições de vida na região não são precárias, e a necessidade de trabalho impede o acompanhamento dos filhos em horário comercial. Em horário compatível, porém, os pais buscam auxiliar os filhos nas atividades escolares.

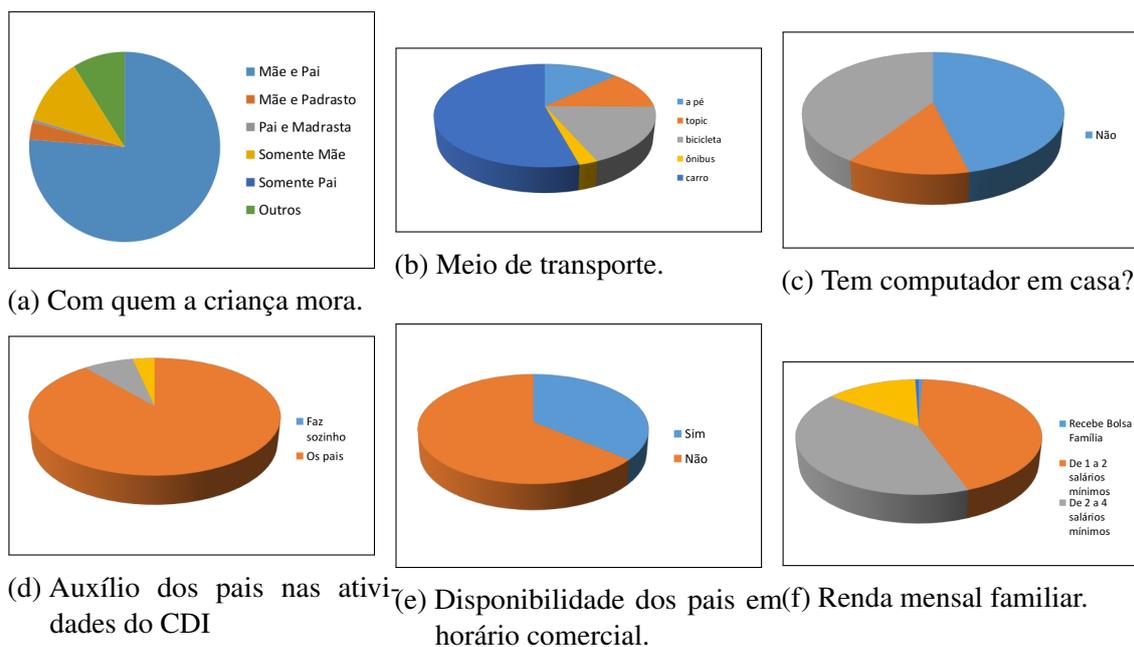


Figura 39. Dados do Projeto Político Pedagógico do CDI (2020).

5.1.3 Proposta Pedagógica para a Educação Infantil

Mesmo com as suas particularidades locais, todos os CDIs de Gaspar buscam seguir uma proposta pedagógica central alinhada com a rede de ensino do municipal. Esse alinhamento se dá por meio da Proposta Pedagógica da Rede Municipal para a Educação Infantil (GASPAR, 2010). Publicada em 2010, após ser construída coletivamente pelos professores da Rede Municipal de Educação Infantil, é um documento que busca nortear o trabalho pedagógico “para e com” as crianças pequenas.

Este “norte” tem duas bases principais. Primeiro, os eixos de Linguagens, Interações e Brincadeiras. O eixo Linguagens visa incrementar as aprendizagens infantis através de atividades que envolvam uso do corpo (linguagem motora); exploração dos sons (linguagem musical); exploração de cores, formas e texturas (linguagem plástica); fala e códigos (linguagem oral e escrita) e noções de espaço, quantidade e números (linguagem matemática). O eixo de Interações diz respeito ao planejamento e organização de situações de contato entre criança-criança, criança-adulto e criança-objeto. Por fim, o eixo das brincadeiras busca aumentar o repertório de interações lúdicas entre as crianças, orientando o educador a observar, coordenar ou integrar-se nas brincadeiras.

A segunda base da proposta é Metodologia de Projetos. Essa seria uma abordagem que permite combinar as intenções pedagógicas do adulto e também estimular a curiosidade da criança. Esse estímulo se dá ao propor projetos em que há investigação de algum tema ou construção de algo com foco em assuntos do mundo infantil (ver Figura 40a). Os temas dos projetos surgem em diálogos

durante rodas de conversa, passeios ou brincadeiras. Partindo do interesse das crianças, os projetos seguem uma estrutura com hipóteses, perguntas de pesquisa, descrições e comparações. Não há prazo de início e fim, e também não há necessidade de trabalhar diariamente com projetos. Quando nenhum projeto está em curso, trabalha-se seguindo os eixos de Linguagens, Interações e Brincadeiras.



(a) Projeto de construção de casa de tijolos (História dos Três Porquinhos).

Fonte: Proposta Pedagógica da Rede Municipal de Educação Infantil de Gaspar (2010).



(b) Projeto para assimilação de quantidades. Turma de 4 a 5 anos.

Fonte: O autor.

A proposta não menciona diretamente o tema das tecnologias digitais. Palavras como *celular* e *computador* não são citadas. Ainda assim, a relação com o “mundo digital” é aderente aos eixos e abordagens metodológicas propostas. As interações, por exemplo, podem ser favorecidas pelo RoPE AR, quando a criança aperta botões e encaixa blocos que provocam reações em um objeto. Há também o uso de uma Linguagem para comunicar uma sequência de ações a um objeto (o brinquedo RoPE). Por fim, a atividade em si é uma Brincadeira, que segundo a proposta deve envolver elementos acolhedores, desafiadores e inclusivos (GASPAR, 2010, p.50).

5.2 PARTICIPANTES DA PESQUISA

Participaram da pesquisa 20 crianças e 3 professoras de um CDI público, além do pesquisador. As crianças têm entre 4 a 6 anos, e frequentam o CDI no período matutino ou vespertino, em semanas alternadas. Quanto às professoras, são mulheres, com formação em Pedagogia e com mais de 10 anos de dedicação à Educação Infantil. Elas declaram ser inábeis para trabalhar com tecnologias digitais “avançadas”, mas usam softwares como editores de texto e navegadores de internet para planejar as atividades com as crianças. O pesquisador é um homem de 26 anos, com formação em Ciência da

Computação e experiência em desenvolvimento de softwares empresariais.

A participação das professoras e crianças se deu em três salas. A primeira sala atende crianças entre 5 e 6 anos (5-6A), enquanto as outras duas atendem crianças entre 4 e 5 anos (4-5A e 4-5B). Cada sala recebe crianças diferentes no período matutino e vespertino.

No primeiro dia da pesquisa, a atividade ocorreu na sala de 5-6A, durante a tarde. No segundo dia a visita ocorreu na sala 4-5A. Por fim, no terceiro dia, a sala visitada foi a 4-5B, tanto de manhã quanto à tarde. Deste modo, quatro grupos de crianças participaram (Quadro 12):

Quadro 12. Encontros e participantes

Grupo	Encontro	Turma	Meninas	Meninos	Professoras (nomes fictícios)
1	Dia 1	5 a 6 anos - Tarde	3	2	Daiana e Maria
2	Dia 2	4 a 5 anos A - Tarde	0	3	Paula e Joana
3	Dia 3	4 a 5 anos B - Manhã	4	2	Vera e Lúcia
4	Dia 3	4 a 5 anos B - Tarde	2	4	Vera e Lúcia
4 grupos	3 dias	4 turmas	9	11	Seis professoras

Fonte: O autor.

A distribuição por sexo ficou balanceada, com 9 meninas e 11 meninos. Já a distribuição por idade se concentrou em 4 e 5 anos (Figura 41), com mediana de 4,58 e média 4,71. Isso é compreensível, pois os experimentos ocorreram em duas salas com crianças de 4 a 5 anos e em apenas uma sala com crianças de 5 a 6 anos.

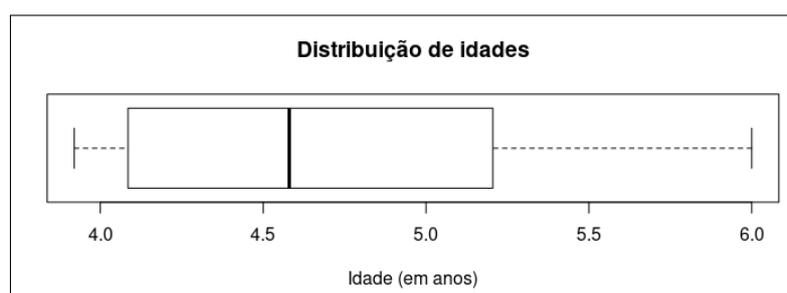


Figura 41. Distribuição de idades.

Fonte: O autor.

Cada sala possui uma professora e uma assistente, denominadas aqui de professoras. Todas são mulheres, e trabalharam de 4 a 28 anos na profissão. Das seis professoras participantes, três responderam diretamente a entrevistas e outras três colaboraram comentando a respeito do projeto ou auxiliando na comunicação com as crianças. Os nomes utilizados no texto são fictícios, para evitar a identificação dos participantes

O recrutamento se deu por conveniência. As crianças participantes frequentavam o ambiente no dia da visita do pesquisador. Não houve distinção de sexo ou presença/ausência de deficiência intelectual. A ordem de participação das crianças ocorreu com o pesquisador solicitando a participação de crianças e as professoras selecionando as mesmas. As crianças que quiseram participar antes da solicitação não foram impedidas. Deste modo, não houve um controle rígido do acesso à interface, mas sim uma organização para evitar mais de três crianças brincando juntas simultaneamente. Quanto às professoras, participaram das entrevistas as responsáveis pela turma em questão ou as que se sentiram confortáveis em responder.

5.3 AMBIENTE

As atividades ocorreram em três salas frequentadas pelas crianças. Em todas as salas o projetor ficou em um suporte, posicionado ao lado de uma fonte de energia e afastado de janelas e portas. As luzes das salas foram apagadas, mas isso não modificou o ambiente a ponto de torná-lo escuro (ver Figura 42). No chão, à frente do projetor, foram fixas duas cartolinas brancas para receber as imagens projetadas. Ao lado do projetor a câmera de um notebook gravou as interações.



Figura 42. Posicionamento do projetor.

As crianças continuaram suas atividades junto às professoras enquanto o pesquisador organizava

os equipamentos. Era perceptível a curiosidade a respeito dos materiais e da figura do pesquisador, tanto pelos olhares quanto por perguntas feitas às professoras. Ao lado da sala da turma 4-5A há um espaço externo gramado. Neste ambiente a professora continuou as atividades no espaço externo, e a programação do RoPE ocorreu na sala com a outra professora presente. As outras duas salas tem espaços amplos, e todas as crianças permaneceram no ambiente durante a atividade. Enquanto duas ou três crianças participavam da pesquisa, as demais continuavam suas atividades em outro ponto da sala.

O CDI tem internet sem fio, porém a qualidade do sinal varia em cada sala. A sala 4-5A apresenta problema de quedas constantes de conexão. Esse problema foi mencionado pelas professoras e ocorreu durante a conexão do projetor na rede local. As demais salas visitadas, porém, tem conexão estável.

5.4 INSTRUMENTOS DE COLETA

A coleta de dados gerou três artefatos: vídeos das interações de crianças com a RoPE AR, entrevistas com as professoras que acompanharam essas interações, e a lista dos algoritmos criados pelas crianças, registrados na CtPuzzle Platform.

5.4.1 Vídeos de interações

A gravação das interações ocorreu logo após a organização do ambiente. Foram gravadas interações de crianças interagindo em duplas, trios ou individualmente, para solucionar desafios de programação e depuração. Um protocolo definiu as etapas e os materiais² a serem usados durante essas interações. Apesar de definir a sequência de ações, o protocolo não impediu flexibilizá-las, de modo a

² Materiais utilizados:

- Projetor (UC46)
- RoPE
- Smartphone Android (Modelos testados: M5c e Redmi 8)
- Duas cartolinas brancas (coladas no chão, receberam a imagem projetada)
- Suporte para o projetor
- Blocos de papelão
- Folha de consentimento
- Conexão com a internet
- Notebook (webcam para gravação dos vídeos)

se adaptarem às sugestões das crianças, dificuldades, curiosidades, e restrições de tempo existentes.

1. Seleção: com aprovação da direção, a seleção dos participantes em cada atividade ocorreu com o pesquisador na sala de aula, perguntando quais crianças gostariam de participar da atividade. As professoras então selecionaram grupos de 2 ou 3, escolhendo as crianças mais próximas umas das outras. Na sala 3 houve alternância de crianças entre os grupos, de modo que algumas entraram e saíram durante a atividade, mas procurando manter um número máximo de 3 crianças simultâneas. Na sala 2 as professoras selecionaram as crianças que consideraram ter mais e menos dificuldades de compreensão. Não foi considerada ou questionada a experiência das crianças com jogos digitais.
2. Ambientação: a ambientação ocorreu como uma conversa inicial entre pesquisador e as crianças, sentados em tapetes emborrachados, próximos ao projetor. O pesquisador perguntou se elas conheciam algum robô. Elas então responderam com frases como “*o meu robô sai luzes*”, ou “*eu lembro daquele robô daquele filme*”. Após isso, as crianças foram instigadas a descrever a aparência e as ações do robô. Por fim, o pesquisador apresentou o RoPE, e falou que a tarefa seria ensiná-lo a caminhar de acordo os desenhos nos blocos de papelão.
3. Reconhecimento dos blocos: para cada um dos cinco blocos, em ordem aleatória, as crianças responderam à pergunta “*o que vocês “acham” que ele [o robô] está fazendo neste desenho aqui?*”. A intenção foi ouvir as respostas, sem julgamentos. Em caso de respostas como “*ele está olhando para o lado*”, o pesquisador perguntou para que lado ele estava “*olhando*”³. Os blocos ficaram ao alcance das crianças, e se observou como as mesmas os manipulavam e se identificavam alguma relação com os botões do RoPE. Buscou-se permitir liberdade durante essas interações: em certa ocasião, por ter blocos repetidos, as crianças entenderam os mesmos como um jogo da memória, o qual foi imediatamente jogado.
4. Programação: o aplicativo e a projeção foram ligados, e as crianças descreveram o que viam na área projetada. O pesquisador explicou às crianças que a tarefa consistia em ajudar o RoPE andar pelo caminho marrom (o caminho mais curto) e pegar a maçã, e para isso deveriam usar os blocos. Para a primeira maçã, o pesquisador explicou o funcionamento, colocando o RoPE na posição inicial, encaixando o bloco Frente no bloco Início e apertando o botão Iniciar do RoPE. As crianças então usaram os blocos e programaram o robô para coletar as próximas quatro maçãs (Quadro 13).

³ Nos desenhos que representam o robô *girando*.

A complexidade da solução das fases é crescente. Enquanto a primeira fase exige apenas um comando, a segunda exige dois. A terceira fase também exige dois comandos, porém acrescenta o comando de giro. Além do número de comandos, a orientação da criança em relação ao RoPE é outro fator que dificulta a resolução. Nas fases 1 e 2 a criança programa o robô estando na mesma orientação que ele. Nas fases 3 a 5, porém, o robô está com seu lado direito voltado para a criança, e ela precisa pensar na perspectiva do brinquedo ou se posicionar para a mesma orientação que o mesmo⁴.

Dúvidas e erros surgiram com frequência durante a programação, e o pesquisador auxiliou fazendo perguntas e dando exemplos quando necessário, e explicando as regras e direções.

5. Depuração: Nesta etapa as crianças foram questionadas a encontrar erros em um algoritmo criado pelo pesquisador. Por quatro vezes, este sequenciou blocos de modo que o brinquedo não coletasse a maçã (Quadro 14). Se, apenas observando a sequência, as crianças não encontraram o erro, o pesquisador solicitou para apertarem o botão Iniciar para ver o robô andar de modo incorreto. As crianças então editaram a sequência de blocos. Nos casos em que a edição do programa ainda não corrigiu o erro, o pesquisador reposicionou o brinquedo no ponto inicial. Enquanto o objetivo da etapa de programação foi a criança *construir* algoritmos, a intenção da fase de depuração foi favorecer a *avaliação* do algoritmo, de modo que as crianças pudessem olhar, discutir, modificar e testar o programa para eliminar o bug e resolver o problema.

Tanto na etapa de programação quanto de depuração não foi exigida a finalização de todas as fases. Nas situações em que todas 1) as fases foram concluídas, 2) a criança perdeu o interesse na atividade, ou 3) ao final do tempo disponível, o pesquisador agradeceu às crianças e finalizou a atividade.

Além de não ser exigida a finalização das tarefas, as trajetórias de cada fase serviram como uma sugestão, e não como uma restrição. Ocorreram modificações em que a criança sugeriu outro ponto de partida ou um caminho mais longo. O aceite ou não destas modificações foi negociado para haver tentativas nos caminhos pré-definidos e também nos caminhos sugeridos pela criança. Da mesma forma, foi permitida a resolução do problema em etapas, quando a criança programou o brinquedo para andar até metade do caminho, por exemplo, e depois programou os demais passos.

⁴ Piaget e Inhelder (1981) citam a dificuldade que crianças têm de imaginar-se em outra perspectiva (egocentrismo).

Quadro 13. Fases de programação

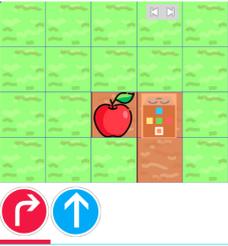
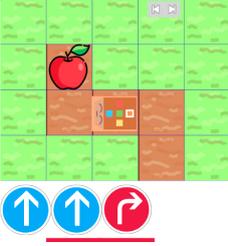
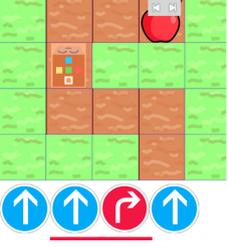
Fase	Descrição	Solução esperada
	<p>Fase 1: Primeira fase e a mais simples. O desafio é andar apenas um passo à frente. A intenção é que a criança compreenda que o robô se move em correspondência ao bloco encaixado, e que o início da execução ocorre ao apertar o botão verde do robô.</p>	Frente
	<p>Fase 2: A ideia aqui é que a criança compreenda que o robô faz mais de um passo por vez. Precisa justapor dois blocos frente para alcançar o objetivo.</p>	Frente, frente
	<p>Fase 3: Nesta fase é adicionada o movimento de giro. A maçã aparece do lado direito do robô, que está virado para frente.</p>	Direita, frente
	<p>Fase 4: A solução desta fase emprega os mesmos comandos que a Fase 2. A diferença é a orientação provável do robô em relação à criança. Se a criança estiver na frente da área projetada, então verá a lateral direita do RoPE. Na Fase 2 criança e robô estão em orientações iguais.</p>	Frente, frente
	<p>Fase 5: Esta é a fase de programação que exige maior número de comandos (três). Além disso a criança precisa imaginar a posição do robô após o giro, para então decidir o comando a ser usado em seguida (frente).</p>	Frente, esquerda, frente

5.4.2 Entrevistas com as professoras

As entrevistas com as professoras ocorreram individualmente, após as atividades com as crianças. O local da entrevista foi o ambiente escolar, enquanto as crianças prosseguiam com outras atividades, acompanhadas por outra professora.

A entrevista ocorreu de forma semi-estruturada (BONI; QUARESMA, 2005), com cinco questões abertas previamente definidas (Quadro 15). O objetivo das perguntas foi explorar o contexto de trabalho do professor, sua visão a respeito da relação entre crianças e tecnologias digitais, e por fim explorar sua visão sobre a atividade proposta com a RoPE AR. Com consentimento das professoras, os áudios das entrevistas foram gravados e posteriormente transcritos (Apêndice F).

Quadro 14. Fases de depuração

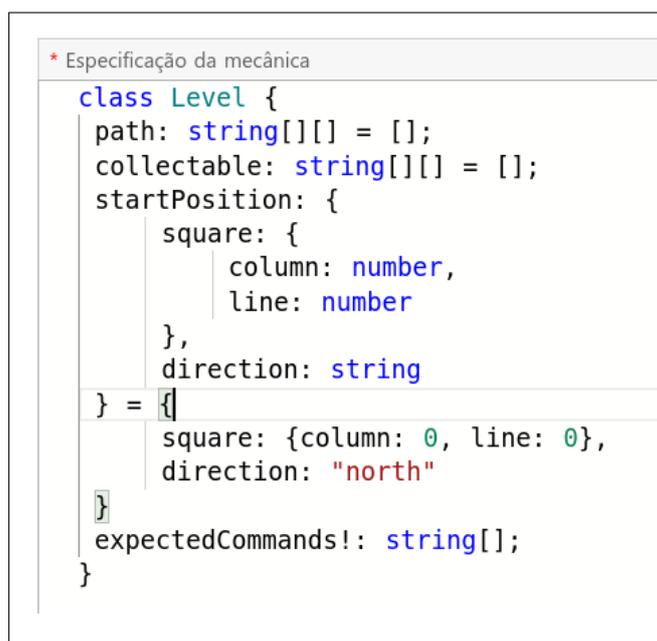
Fase	Descrição	Solução esperada	Tipo de erro
	Fase 6: A maçã está logo à frente, porém o algoritmo tem um giro à esquerda.	Frente	Comando incorreto
	Fase 7: Nesta fase o erro é o comando de giro. O robô precisa girar à esquerda, mas o comando utilizado é o de girar à direita.	Esquerda, frente	Comando incorreto
	Fase 8: O erro nesta fase está na ordem dos comandos utilizados. Se o brinquedo avança duas vezes ele passa da posição onde deve girar à direita. A tarefa consiste em reordenar os comandos já existentes, em vez de adicionar ou remover comandos.	Frente, direita, frente	Ordem dos comandos
	Fase 9: Tarefa de maior dificuldade em todas as etapas, pois exige quatro comandos. Assim como na fase 8, o erro está na ordem dos comandos utilizados.	Frente, direita, frente, frente	Ordem dos comandos

Quadro 15. Questionário com as professoras

Pergunta	Motivação
<i>Q0 — Você poderia falar sobre suas experiências no trabalho voltado à Educação Infantil?</i>	Compreender as experiências passadas e o contexto de trabalho da professora.
<i>Q1 — O que você pensa sobre o contato de crianças com tecnologias digitais?</i>	Identificar sentimentos do professor em relação à tecnologia e também identificar os dispositivos citados.
<i>Q2 — Que importância você daria para a visibilidade no aprendizado das crianças?</i>	Verificar se a opinião das professoras se alinha com a proposta da interface.
<i>Q3 — Você poderia pensar em alguma atividade usando os mesmos princípios propostos na atividade de hoje, seguindo a ideia de projetar imagens no chão e ter essa interação com um robô?</i>	Compreender a percepção do professor sobre a aplicabilidade da proposta.
<i>Q4 — Que problemas você percebe na proposta? Quais aspectos poderiam melhorar?</i>	Identificar falhas da interface na visão do professor.

5.4.3 Registro de respostas na CtPuzzle Platform

Como descrito na seção Subseção 4.2.1.4, a CtPuzzle Platform é uma plataforma que possibilita configurar mecânicas de puzzles para criar testes de avaliação do PC. A RoPE AR representa uma mecânica com diferentes níveis de dificuldade. Em cada nível há um mapa, um ponto inicial, e um caminho a ser percorrido (Figura 43).



```

* Especificação da mecânica
class Level {
  path: string[][] = [];
  collectable: string[][] = [];
  startPosition: {
    square: {
      column: number,
      line: number
    },
    direction: string
  } = {
    square: {column: 0, line: 0},
    direction: "north"
  }
  expectedCommands!: string[];
}

```

Figura 43. Especificação da mecânica.

Fonte: O autor.

Após definir os atributos da mecânica, a plataforma possibilita configurar itens com estes atributos. Estes itens são agrupados em testes. As fases de programação e depuração (Quadro 13 e Quadro 14) compõem um teste de 9 itens.

Por fim, cada item tem uma *resposta esperada*. No caso da RoPE AR, a resposta esperada, para cada fase, é formada por uma sequência de comandos correspondentes a movimentos que o RoPE precisa fazer para percorrer um caminho. Para cada um dos 9 itens, a comparação entre a resposta esperada e a *resposta dada* produz um *score*. Ele representa a assertividade na solução do problema. Essa comparação se dá numa função de cálculo de score. A Figura 44 mostra a fase 1 configurada na plataforma e instanciada pela RoPE AR: a resposta esperada é um avanço à frente.

Ressalta-se que, durante a resolução do desafio, é comum a criança realizar mais que uma tentativa. Em cada tentativa ela insere os comandos e pressiona o botão iniciar. A cada início de execução a lista de comandos é armazenada em uma lista de tentativas. Essa lista de comandos usados

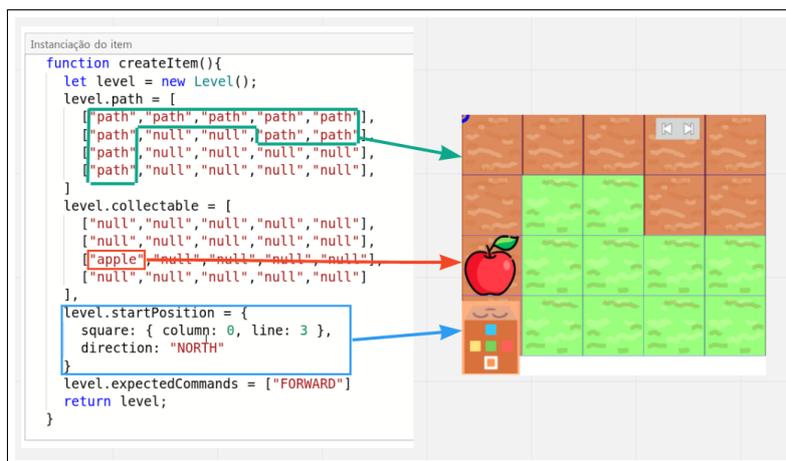


Figura 44. Item configurado na plataforma e instanciado no aplicativo RoPE AR.
Fonte: O autor.

nas tentativas é utilizada no cálculo do escore.

5.4.4 Métodos de análise dos dados

A análise dos dados se deu pelos métodos indutivo e dedutivo. Na análise indutiva os vídeos foram assistidos repetidas vezes, sem uma pré-definição de códigos. Nessa etapa o software Qualcoder, um software do tipo *Computer Assisted Qualitative Data Analysis Software* - Software de Apoio à Análise de Dados Qualitativos (CAQDAS), gratuito e de código aberto, serviu como ferramenta de codificação. Em um processo iterativo, foram marcados trechos nos quais as crianças transitaram de estados de repouso para estados ativos, considerando suas gesticulações e falas. No decorrer dos vídeos surgiram novos códigos, conforme o olhar do pesquisador se voltou para aspectos que surgiram em cada filmagem. Entre os aspectos observados, o Qualcoder permitiu marcar em azul as interações com botões, laranja as interações com blocos e em roxo momentos em que houve depuração (Figura 45). Momentos em que houve mediação de adultos foram marcados em amarelo. Com isso, a partir do terceiro vídeo (Figura 46), aparecem trechos marcados em amarelo, pois neste caso as professoras interagiram diretamente durante a brincadeira. Por fim, os trechos em verde-escuro, lilás e verde-claro marcam os trechos de apresentação da tarefa, a etapa de programação, e a etapa de depuração.

Esse processo de “codificação aberta” permitiu identificar códigos e selecionar as categorias que apareceram com mais frequência. As categorias identificadas foram *Interação com os botões*, *Interação com os blocos*, *Interação entre crianças*, *Depuração*, *Estratégias de solução de problema*, *Percepção de realidade aumentada* e *Mediações de um adulto*. Como foco do trabalho, foram analisadas as categorias relacionadas a interação com os blocos, realidade aumentada e depuração.

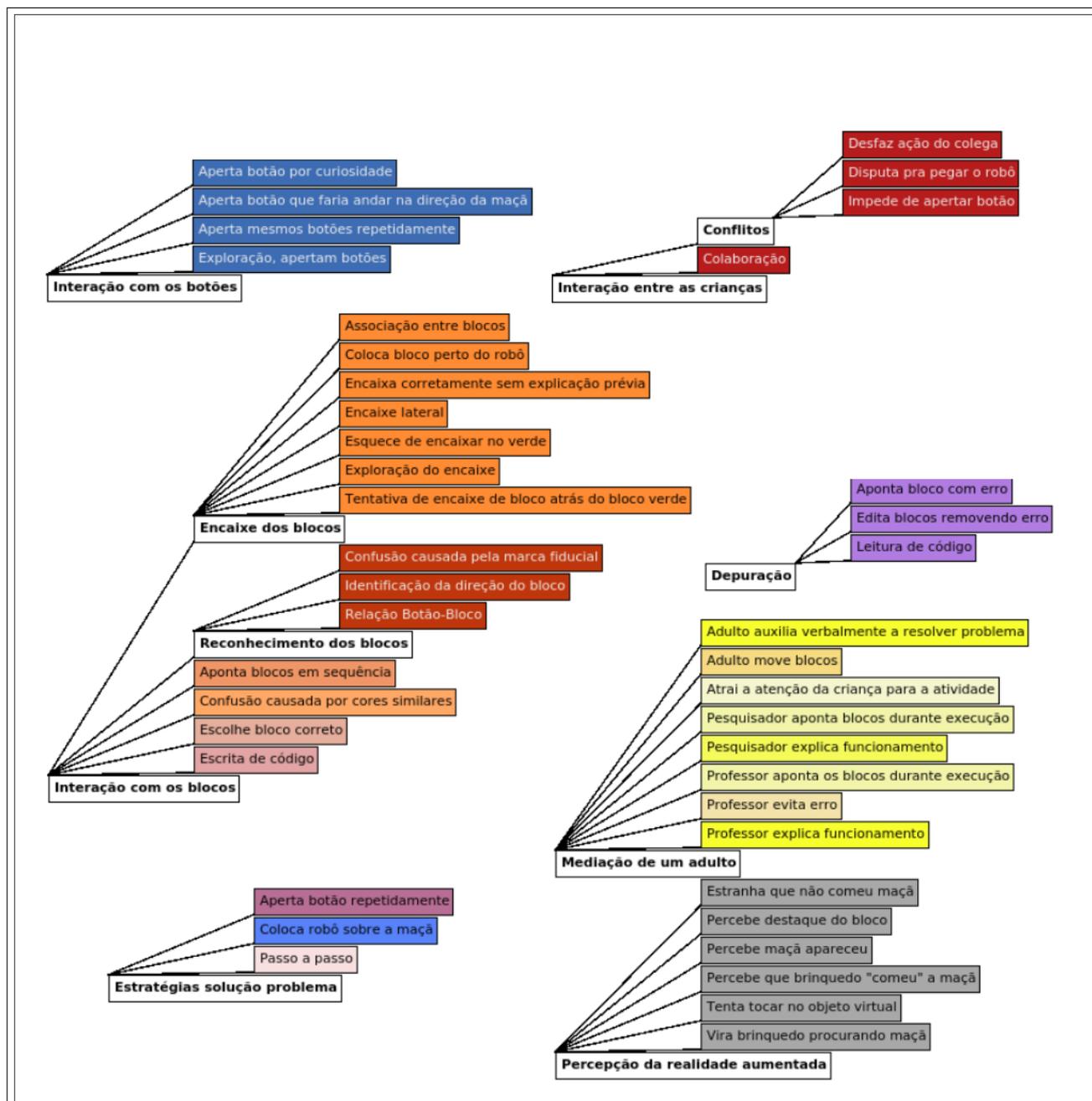


Figura 45. Processo de categorização.
Fonte: O autor.

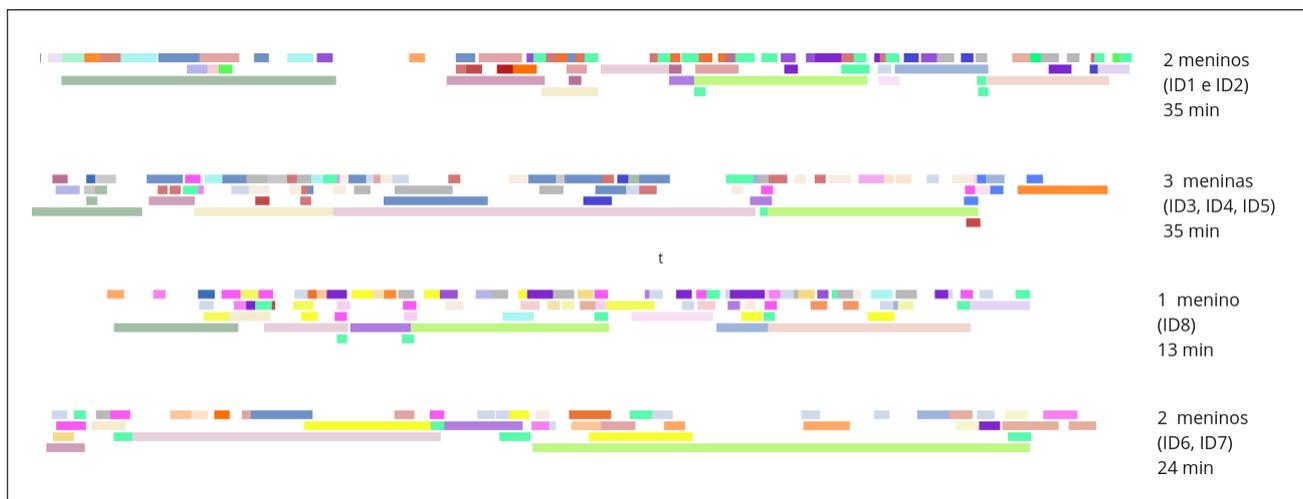


Figura 46. Marcações de trechos dos vídeos em cores que indicam seu código.

Fonte: O autor.

A análise dedutiva tem como base o método de análise das filmagens seguido por Santana (2015) e Blikstein (2014). Esses autores analisam vídeos em que estudantes resolvem desafios de montagem de artefatos físicos, como um brinquedo programável ou uma estrutura em palitos. Nestes trabalhos as ações dos estudantes são codificadas de acordo com uma máquina de estados, em que os estudantes podem estar no estados como Modificar, Planejar, Executar, Refazer e Repousar. Esta pesquisa se apoia no mesmo método por entender que a atividade proposta às crianças envolve a mesma combinação de manipulações físicas e raciocínio lógico voltado à resolução de um problema.

5.5 CONSIDERAÇÕES

Este capítulo apresentou o ambiente, os participantes, e a coleta de dados da interação com uma interface tangível de programação em blocos, associada a realidade aumentada projetiva. Em resumo, os dados obtidos são vídeos, entrevistas e registros de log, obtidos durante três visitas a um centro de desenvolvimento infantil. O número de participações não é significativo numericamente⁵, mas a gravação de vídeos totalizou 4,5 horas de dados passíveis de análise qualitativa, descrita no próximo capítulo.

⁵ Participação direta de 20 crianças e 3 professoras.

6 RESULTADOS

O Quadro 16 apresenta as equipes que participaram dos experimentos. As equipes 1 e 2 são da turma de 5 a 6 anos, e as demais são das turmas de 4 a 5 anos. O tempo de participação de cada equipe variou entre 13 e 35 minutos, devido a casos de não finalização da atividade. O tempo de 35 minutos, porém, foi suficiente para concluir todas as etapas, desde a ambientação até os problemas de depuração. Outro fator de influência na duração da atividade foi a troca de participantes, que entraram na brincadeira depois do início ou saíram antes do fim. Nesses casos, a equipe da criança é o grupo onde ela permaneceu por mais tempo. A criança C10 também integrou duas equipes para C14 não brincar individualmente.

Quadro 16. Equipes.

Equipe/Tempo	Criança	Idade	Sexo
Equipe 1. 35min	C1	5:10	Masculino
	C2	5:7	Masculino
Equipe 2. 35min	C3	5:10	Feminino
	C4	5:11	Feminino
	C5	6:0	Feminino
Equipe 3. 24min	C6	4:10	Masculino
	C7	3:11	Masculino
Equipe 4. 13min	C8	4:3	Masculino
Equipe 5. 23min	C9	4:7	Masculino
	C10	4:8	Masculino
	C11	4:8	Masculino
Equipe 6. 19min	C12	4:2	Feminino
	C13	4:3	Feminino
Equipe 7. 10min	C14	4:10	Feminino
	C10	4:8	Feminino
Equipe 8. 30min	C15	4:7	Masculino
	C16	4:5	Feminino
Equipe 9. 20min	C17	4:3	Feminino
	C18	4:5	Masculino
Equipe 10. 22min	C19	4:4	Masculino
	C20	4:6	Masculino

6.1 ANÁLISE INDUTIVA

Essa seção apresenta as observações a respeito de como as crianças perceberam e interagiram com os blocos de código tangíveis, e em seguida comenta as suas percepções sobre os elementos de

realidade aumentada.

6.1.1 Interação com os blocos

Após as etapas de seleção e ambientação, as crianças iniciaram o reconhecimento dos blocos. O pesquisador mostrou um bloco por vez, selecionado aleatoriamente, para as crianças dizerem seu significado. Essa etapa é importante, pois a programação e a depuração dos algoritmos dependem de compreender o significado dos blocos.

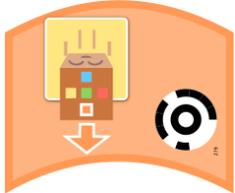
De modo geral, as crianças entenderam as direções dos blocos, e demonstraram isso principalmente por gestos com as mãos. O principal erro foi confundir as setas de “frente” e “trás” com “cima” e “baixo”. As respostas gestuais indicaram mais erros de vocabulário do que de interpretação. Em 9 momentos as crianças usaram os termos “cima” ou “baixo”, mas apontando para “frente” ou “trás”, e apenas 1 criança realmente entendeu essas setas como sinais verticais. Para os blocos de “esquerda” e “direita”, as crianças usaram o termo “pro lado”, enquanto apontavam para a direção correta (Quadro 17).

A associação dos blocos com os botões do RoPE também ocorreu naturalmente, sem a solicitação do pesquisador. Ao perceber que a similaridade de cores entre os blocos e os botões, C1 e C2, e também C10 e C11, apontaram para os botões e aproximaram os blocos dos mesmos. Um caso diverso apareceu na Equipe 2, em que as crianças C4, C5 e C6 ignoraram os blocos e mantiveram atenção apenas nos botões do brinquedo.

Essas interações também evidenciaram a necessidade de modificar o design dos blocos. Na segunda fala do Quadro 17, C4 aponta o círculo da marca fiducial do bloco Frente e entende que o brinquedo está “rodando”. A criança entendeu o bloco posteriormente, mas sua primeira percepção indica que a marca fiducial chamou sua atenção. Portanto, o ideal seria eliminar a marca, pois ela não possui um significado útil para a criança. Em outro caso, C17 falou sobre o bloco Frente: “*Ele “tá” secando as mãos*”. Uma possível origem desta resposta seria entender os quatro riscos desenhados no bloco Frente como símbolo de “vento”. Essa criança compreendeu o significado correto posteriormente, mas sua fala indica que a imaginação fértil da criança, associada a detalhes de design, a fizeram obter uma conclusão inesperada.

O Quadro 18 apresenta exemplos das respostas obtidas. Destaca-se a primeira fala, de C2, que ao analisar o bloco Frente diz: “*ele está botando... o sistema dele pra cima*”. Provavelmente a criança não sabe o que é um sistema, mas sabe que é algo associado a um robô. A dinâmica da atividade

Quadro 17. Falas sobre os símbolos dos blocos.

Bloco	Falas e diálogos
	<p>C2: “Ele tá botando... o sistema dele pra cima.” C1: “Ele tá olhando pra cima.”</p> <p>C4: “Rodando” [Aponta marca fiducial, que é redonda]</p> <p>C4 e C5: “Pra aquele (lado)” [apontam para frente] C3: “Pra aquele” [aponta pra cima] Pesquisador: “Pra cima?” [C3 confirma com a cabeça]</p> <p>C10: “Uma flechinha” Pesquisador: “Uma flechinha pra onde?” C10: “Pra cima” [aponta pra frente] C9: “Tipo daquelas ferramentas” [aponta brinquedos à frente]</p> <p>Pesquisador: “O que ele tá fazendo aqui nesse desenho?” C13: “Dormindo!”</p> <p>C17: “Eu acho que ele tá secando as mãos... Eu acho que ele tá indo pra frente”</p> <p>Pesquisador: “Esse bloquinho ele tá indo pra onde?” C15: “Pra pegar a maçã.”</p>
	<p>C1: “Esse aqui ele tá olhando pra baixo.”</p> <p>C10: “Pra baixo.” C9: “Pra baixo!”</p> <p>Pesquisador: “E onde é pra baixo aqui no robô?” C10: “Gramá” [Aponta para o chão] Pesquisador: “Qual é o botão que faz ele ir pra trás?” C9: “Aqui” [aponta botão trás]</p> <p>C17: “Ele tá indo pra trás” C5: “Pra baixo.” C4: “Esse aqui ele tá olhando pra baixo.”</p>
	<p>C2: “Ele tá olhando pra aquele lado.” [aponta para a esquerda] C5: “[tá rodando] pra aquele [lado].” [gira corpo para a esquerda] C10: “Esse aqui ó.” [aponta para a esquerda]</p>
	<p>C2: “Esse aí ele tá olhando pro lado.” [aponta para a direita] C10: “Pro lado.” [aponta para botão do RoPE]</p> <p>Pesquisador: “Ele tá fazendo alguma coisa?” C13: “Tá dormindo! De novo.”</p>
	<p>C1: “Esse aqui também [está olhando pra cima].” C1: “Esse aqui é pra tirar vários.”</p>

Quadro 18. Reconhecimento dos blocos

Categoria	Descrição	Frequência
"pro lado"	Aponta para a direção correta, mesmo não usando as palavras “ <i>direita</i> ” ou “ <i>esquerda</i> ”	5
"pra cima"	Fala ou aponta direção correta, mesmo falando outras palavras	4
Outros	Menciona algo não relacionado a movimento	3
"pra baixo"	Aponta ou fala a direção correta usando outras palavras	2
Relaciona com o botão	Aponta botão relacionado	2
Menciona esquerda/direita	Menciona palavras “ <i>esquerda</i> ” e “ <i>direita</i> ” durante reconhecimento dos blocos	1
"pra trás"	Fala a palavra “ <i>trás</i> ”	1
Rodando	Identifica botão frente ou trás como rodando	1
Aponta pra cima	Confunde frente com cima, apontando pra cima	1
Pro chão	Diz que o brinquedo está indo pro chão, sem indicar a direção “ <i>trás</i> ”	1
Indo pegar a maçã	Diz que o brinquedo está “ <i>Indo pegar a maçã</i> ”	1

Fonte: O autor.

não permitiu investigar, naquele instante, qual o entendimento da criança sobre o tema, mas temas relacionados à computação apareceram posteriormente:

[C2 Clica diversas vezes em todos os botões]

C1: “*Para, senão ele vai bugar!*”

Pesquisador: “*Bugar? o que é bugar?*”

C1: “*É quando ele faz assim ó*” [C1 e C2 fazem posição de “estátua”.]

Esta criança, portanto, entende que *bugar* é algo relacionado a travamentos e deve ser evitado. Outras crianças apresentaram outras justificativas quando o robô não se moveu: C14 disse: “*Não anda esse robô. Tava dormindo, coitadinho*”. O vocabulário “avançado” da Equipe 1 não apareceu nas outras equipes.

A Equipe 1, além de compreender o significado dos blocos, iniciou a exploração dos mesmos, relacionando-os uns com os outros. As crianças começaram a encaixar os blocos iguais entre si, formando 4 grupos de blocos (4 Frente, 2 Esquerda, 2 Direita, 2 Trás) (Figura 47a). O pesquisador questionou o motivo de tal agrupamento, conforme o diálogo a seguir:

[Crianças encaixam os blocos]

Pesq: “*Dá pra montar né? Dá pra encaixar*” 01.59

[C2 tenta encaixar peças de cores diferentes]

C1: “*Não, esse aqui não pode. Esse aqui não é desse.*” 02.06

C2: “*É um jogo...*” 02.08

C1: “*Esse aqui tá olhando pra baixo.*” 02.11 [segura bloco Trás]

C1: “*Esse aqui vou conectar com esse, e esse conecta com esse*” 02.20 [unindo peças de mesma cor]

Pesq: “*Porque esses assim?*” 02.30

C1: “*Porque tão olhando pro mesmo lado né!*” 02.32

Pesq: “*Ah... e pela cor né*” 02.36

C1: “*Não, porque tão olhando pro mesmo lado!*” 02.40

Pesq: “*Ah... entendi...*” 02.40

A justificativa para tal associação, portanto, foi a posição do brinquedo desenhado, e não cor do bloco. Percebendo a similaridade entre pares de blocos, C1 sugeriu que poderia ser jogado o "Jogo da Memória". O pesquisador e os dois meninos então viraram os blocos com a face para baixo (Figura 47b), e, cada um por vez, virou duas peças.



(a) Relacionamento entre blocos



(b) Jogo da memória, proposto pela dupla

Fonte: O autor.

Outro aspecto de design a ser melhorado apareceu durante o jogo: os blocos Direita e Trás tem cores muito parecidas (tons de vermelho e laranja). Ao procurar blocos iguais, C2 virou um bloco Trás e um bloco Direita e considerou um ponto ganho. C1 alertou o colega ao perceber o engano, e C2 devolveu os blocos. O diálogo revela a interação entre as crianças durante o jogo:

C2: “*Agora é minha vez.*” [vira peças vermelha e laranja, e pensa que acertou] 11.57

C1: “*Errou, é diferente.*” 12.00

Pesq: “*É diferente né. Mas é parecido, é parecido.*” 12.05

A mesma situação ocorreu com C8, que selecionou o bloco Trás ao tentar fazer o brinquedo girar à Direita. A Prof_6 então atuou direcionando a atenção da criança para ela perceber o engano:

Pesq: *“Viu que ele foi pra trás? Porque aqui tá o bloquinho laranja. O que ele tem que fazer?”*

C8: *“Vermelho!”*

Pesq: *“E qual é o bloquinho vermelho?”*

C8: *“Esse aqui ô”* [pega o laranja]

Prof_6: *“Esse é vermelho, tem certeza? Olha bem...”*

Pesq: *“Qual é mais vermelho. Esse ou esse?”* [coloca blocos esquerda e trás lado a lado]

Prof_6: *“Olha a flechinha se é igual.”*

C8: *“Vermelho é esse.”* [aponta bloco Direita]

Esses dois diálogos demonstram que a interação com os objetos viabilizou conversas entre as crianças e de professor para criança. Nos dois casos uma criança foi alertada por colega ou professor sobre um erro, e pode então desfazê-lo manipulando os blocos de papelão. No primeiro caso a correção do erro consistiu em virar a face dos blocos para baixo, pois não formavam um par do jogo da memória. No segundo caso, a correção consistiu em encaixar outro bloco na sequência de blocos. Isso demonstra que ações similares surgiram tanto na brincadeira sugerida pelas crianças quanto na construção de um algoritmo. São ações naturais, presentes nas brincadeiras, mas que viabilizam o contato com conceitos de computação e PC.

6.1.2 Percepções de realidade aumentada

O diferencial da RoPE AR para outros tipos de interface de programação é a união entre realidade aumentada e blocos tangíveis. Neste sentido, a questão é: como as crianças perceberam e interagiram com a RA?

Para respondê-la, a análise dos vídeos apontou alguns tipos de eventos em que as crianças perceberam os elementos projetados e os relacionaram com objetos reais. Para contar esses eventos, foram considerados gestos, falas e reações de animação ou estranhamento das crianças (Quadro 19).

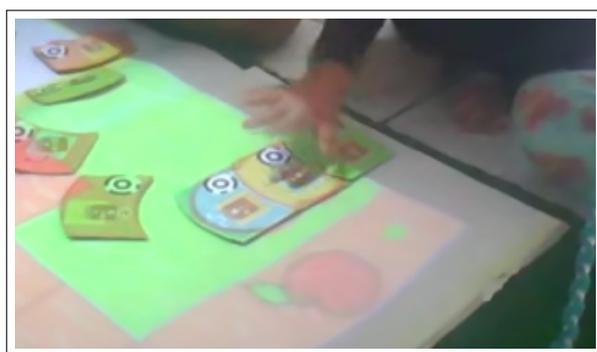
O evento mais frequente (32) foi as crianças apontarem para um objeto projetado, na maior parte dos casos a maçã. Ela era a figura principal do mapa projetado, a ser capturada pelo RoPE. Portanto, o fato das crianças apontarem para ela apenas indica que ela estava suficientemente visível.

Quadro 19. Eventos de percepção de realidade aumentada

Categoria	Descrição	Frequência
Aponta para figura projetada	Criança aponta para uma figura projetada	32
Percebe que brinquedo "comeu" a maçã	Criança fala ou gesticula indicando que o brinquedo "comeu" a maçã	20
Levanta brinquedo, procurando maçã	Criança levanta o brinquedo procurando a maçã que desapareceu	13
Percebe que maçã apareceu	Criança aponta para a maçã que acabou de parecer	10
Estranha que não comeu maçã	Criança estranha quando o brinquedo não come a maçã ao chegar no quadrado dela	5
Percebe destaque do bloco	Criança aponta o bloco destacado pelo projetor	3
Tenta tocar no objeto virtual	Criança coloca a mão sobre a imagem de alguma figura projetada	3
Coloca bloco sobre a maçã	Criança percebe a localização da figura projetada e coloca um objeto físico sobre ela	2
Percebe um detalhe de uma figura projetada	Menciona algum detalhe, como a folha da maçã	1

Fonte: O autor.

O evento a ser destacado, porém, foi as crianças levantarem o brinquedo RoPE e o virarem de “cabeça para baixo” após ele “comer a maçã”. Esse evento ocorreu por 13 vezes, em 5 equipes. Ele indica que as crianças tentaram entender se o destino da “maçã” foi a “barriga” do robô. Uma das crianças também olhou ao redor do mapa e procurou a maçã embaixo dos blocos de papelão. Esses eventos evidenciam que houve dificuldade em distinguir os objetos reais dos virtuais.



(a) C11 coloca a mão sobre os blocos



(b) C10 encaixa blocos para ver a iluminação

Figura 48. Exploração dos blocos

Outros exemplos de interação com a RA foram tentativas de segurar a maçã ou tocar nos círculos de destaque dos blocos. A Equipe 5, por exemplo, percebeu o destaque dos blocos que aparece quando um algoritmo válido é criado. No primeiro momento em que a equipe, de modo exploratório, criou um algoritmo e a RoPE AR o destacou, C11 percebeu o destaque e aproximou a mão dos blocos (Figura 48a). Em seguida, a colega C10 entendeu que o encaixe dos blocos provocou sua

iluminação. Como consequência, C10 criou uma longa sequência de blocos, interessada em ver o efeito da iluminação sobre os mesmos (Figura 48b).

Essas ações indicam que a associação de elementos virtuais aos reais foi compreendida. Entretanto, essas ações ocorreram esporadicamente e as crianças rapidamente perderam o interesse ao perceber que os objetos virtuais não eram manipuláveis diretamente. As crianças também não demonstraram interesse pelo destaque, projetado em azul sobre blocos durante a execução. Em apenas três momentos houve apontamentos os círculos de destaque e nenhuma fala sobre eles. Esse recurso serviu como um auxílio para o pesquisador explicar a sequência de execução dos blocos, mas não eliminou a necessidade de apontá-los a mão.

Por fim, as crianças apresentaram interesse pela figura da maçã, que é uma entidade conhecida¹. Havia forte desejo de fazer o robô capturá-la. Para isso, as principais estratégias foram apertar os botões, mesmo que desativados, testar a combinação de diferentes blocos, e também empurrar o brinquedo com as mãos. O alcance do objetivo provocou reações de espanto e animação. Neste sentido, apesar de a tarefa ser repetitiva (sempre capturar a maçã), não se percebeu perda de engajamento. Pelo contrário, as crianças quiseram continuar a brincadeira mesmo após o término.

6.2 ANÁLISE DEDUTIVA

A análise dedutiva teve como objetivo comparar as equipes e as atividades de programação e depuração. Ela segue a proposta de Blikstein (2014), e inicia com uma definição prévia de códigos representando as ações das crianças. O Quadro 20 demonstra as ações mapeadas. Esses códigos foram analisados por dois pesquisadores de modo a obter um consenso sobre quais utilizar na etapa final de codificação. Ao obter consenso, o pesquisador codificou as etapas de programação e depuração captadas nos vídeos.

As ações foram então agrupadas em estados de alto nível (Quadro 21): entender, programar, depurar e executar. O estado Entender agrupa ações em que a criança busca informações para prosseguir a resolução da atividade, ou apresenta algum sinal de avaliação de um resultado. Programar agrupa ações de modificação do algoritmo do RoPE, seja por botões ou blocos. Depurar também abrange essas ações, mas quando ligadas a um erro recém-ocorrido. Por fim, o estado Executar se refere a iniciar execução e o estado Repousar significa apresentar sinais de distração.

¹ Maçã é uma fruta presente nas refeições das crianças no CDI.

Quadro 20. Ações mapeadas

Código	Descrição
Adiciona blocos	Aproxima dois ou mais blocos e os encaixa corretamente
Adiciona bloco para resolver erro	Escolhe um bloco específico após a ocorrência de um erro
Analisa robô	Manipula robô olhando para diversas partes dele
Aperta botões	Aperta botões direcionais do brinquedo
Aponta bloco	Aponta um bloco na lista de blocos
Aponta bloco com erro	Aponta um bloco como a causa de um erro
Aponta botão	Indica qual botão corresponde a uma ação
Aponta caminho	Com a mão, mostra a direção que o RoPE deve seguir
Aponta maçã	Indica a posição da maçã
Aproxima bloco do robô	Aproxima um bloco do robô, com o intuito de que ele faça a ação do bloco
Coloca brinquedo no início	Após erro, coloca brinquedo na posição inicial
Coloca brinquedo sobre a maçã	Move diretamente o brinquedo para a posição final
Usa comandos de voz	Fala a ação desejada para o RoPE
Comemora	Apresenta sinais de alegria imediatamente após uma ação do brinquedo
Cria hipótese	Sugere uma ação para solucionar um problema
Distrai	Olha pros lados ou brinca com outros brinquedos da sala
Escuta	Olha para o pesquisador ou professora enquanto eles falam
Executa algoritmo com erro	Ciente do erro, mas inicia a execução para testar
Executa passo	Aperta no botão que indica próxima ação do brinquedo, no modo de execução passo a passo
Fala que comeu maçã	Percebe e fala que brinquedo capturou a maçã
Fala sobre erro	Comenta sobre uma ação errada do RoPE
Gira bloco	Gira um bloco para seta apontar em outra direção, mesmo que não encaixe na sequência de blocos
Gira robô	Rotaciona brinquedo para que fique de frente para a maçã
Inicia execução	Aperta botão Iniciar
Le algoritmo	Aponta, um por vez, os blocos encaixados em sequência
Levanta robô procurando maçã	Após maçã sumir, levanta brinquedo do mapa
Observa	Olha para blocos ou robô, sem tocar em nada
Pensa com objeto na mão	Segura objeto, olha para ele e para o mapa
Percebe que maçã apareceu	Aponta para maçã após ela aparecer
Remove blocos	Retira um ou mais blocos de uma sequência de blocos encaixados
Remove bloco com erro	Retira um ou mais blocos de uma sequência após a execução de um erro
Reordena blocos	Muda a posição de um ou mais blocos na sequência de blocos
Seleciona blocos	Procura um bloco para encaixar na sequência de blocos
Simula passos do robô	Com a mão, move o RoPE e olha para os blocos
Substitui bloco	Retira bloco e adiciona outro em seu lugar
Substitui bloco com erro	Retira bloco e adiciona outro em seu lugar após um erro

Fonte: O autor.

Quadro 21. Estados

Estados	Códigos
Entender	<ul style="list-style-type: none"> • Analisa robô • Aponta bloco, botão, caminho ou maçã • Coloca brinquedo sobre maçã • Observa, olha blocos, olha robô • Comemora, cria hipótese, • Gira bloco ou robô • Escuta • Fala que "comeu" maçã, percebe que maçã apareceu
Programar	<ul style="list-style-type: none"> • Adiciona bloco, aperta botões • Aproxima bloco do robo • Comandos de voz • Remove bloco • Organiza blocos, reordena blocos, substitui bloco • Seleciona blocos
Depurar	<ul style="list-style-type: none"> • Adiciona bloco resolvendo erro • Aponta bloco com erro • Coloca brinquedo no inicio após erro • Executa algoritmo com erro • Executa passo depuração • Fala sobre erro • Remove bloco com erro, substitui bloco com erro
Executar	<ul style="list-style-type: none"> • Inicia execução
Repousar	<ul style="list-style-type: none"> • Se distrai

Fonte: O autor.

A Figura 49 demonstra o percentual de ações para todas as equipes nas fases de programação. Nessas fases a criança inicia construindo um algoritmo para resolver um problema, e podem ocorrer erros ou não. Nesse contexto, as ações mais frequentes foram entender (46.67%), seguidas de programar (30.26%) e executar (19.47%). Ações relacionadas a depuração foram bem menos frequentes, com apenas 1.84%. Por fim, apenas 1.75% das ações estão relacionadas a repouso. Portanto, com maior

frequência as crianças observaram e tentaram modificar os algoritmos, mas sem perceber ou corrigir erros. Também permaneceram focadas na tarefa.

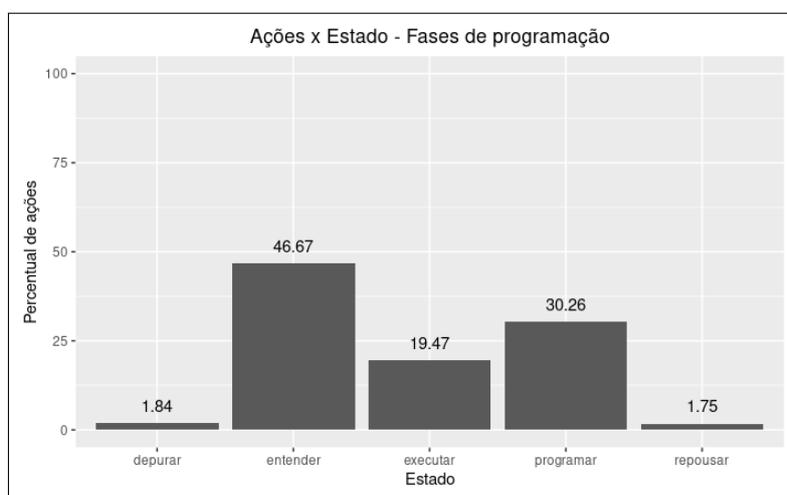


Figura 49. Percentual de ações em cada estado para todas as equipes, somente nas fases de programação.

Fonte: O autor.

A Figura 50 mostra o percentual de ações, em cada estado, somente nas etapas de depuração. O número de ações ligadas a depurar algoritmos aumenta para 15.58%. Já as ações de entender e executar permanecem iguais aos níveis anteriores com margem de 2%. As ações de programar diminuem para 16.3%. Ocorreu, portanto, uma transferência do estado de programar para o estado de depurar.

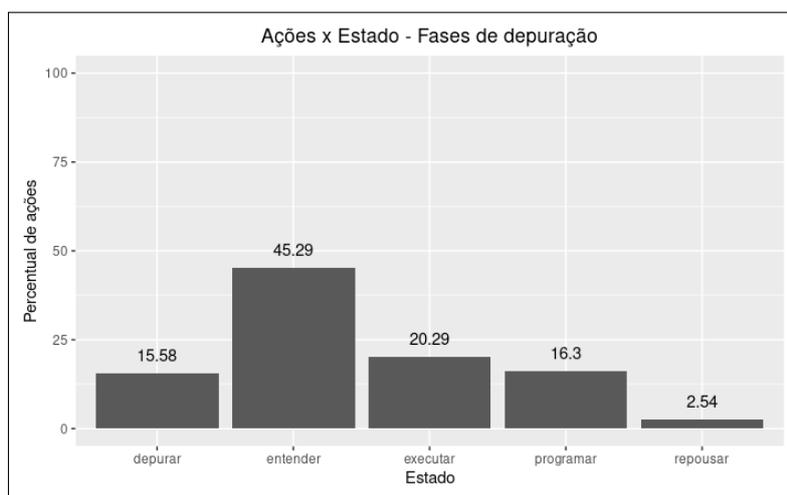


Figura 50. Percentual de ações em cada estado para todas as equipes, nas fases de depuração

Fonte: O autor.

6.2.1 Tempos por fase

Além da codificação dos estados, também foram analisados os tempos despendidos por cada equipe em cada uma das 9 fases propostas. A obtenção dos tempos se deu por consultas SQL ao banco de dados armazenado pelo software Qualcoder. Após a extração, esses dados foram conferidos manualmente de modo a associar os tempos de cada fase às respectivas equipes. Por fim, a análise se deu utilizando a linguagem R no programa RStudio. Ressalta-se que não houve um critério pré-definido para a análise desses tempos. O objetivo foi explorar as diferenças e entre os tempos entre as equipes e entre as tarefas.

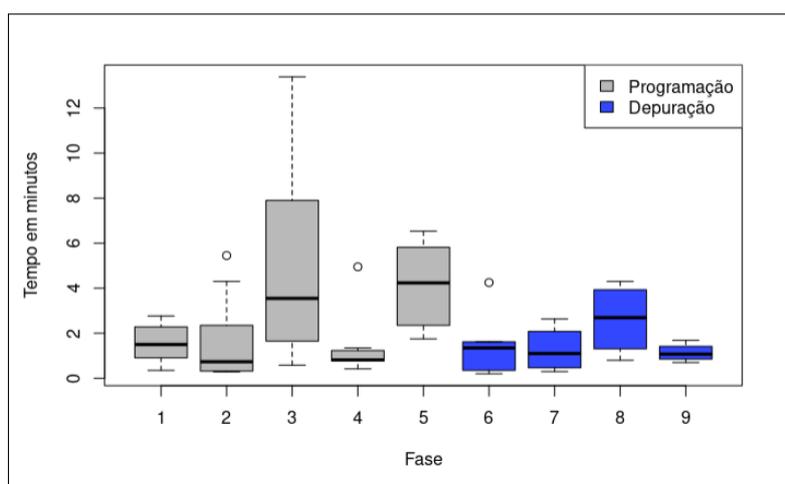


Figura 51. Distribuição de tempo por fases.

Fonte: O autor.

A Figura 51 mostra os tempos de resolução de cada uma das 9 fases. Em cinza aparecem as 5 fases de programação, e em azul as 4 fases de depuração. Na etapa de programação, as fases 3 e 5 exigiram os maiores tempos de resolução. Uma possível causa para isso é que essas fases utilizam comandos de giro, que, conforme a literatura (PIAGET, 1964), representa um obstáculo para as crianças. Destaca-se que a fase 3 é a primeira com comandos de giro, apresentou o maior tempo (mediana de 4 minutos).

A fase 7, de depuração, porém, é um contraponto. Ela também usa um comando de giro, sendo um espelho da fase 3. Enquanto uma usa os comandos Direita e Frente, a outra é solucionada por Esquerda e Frente. Teriam, portanto, a mesma complexidade. Apesar disso, a fase 7 foi resolvida em menos da metade do tempo da fase 3. Alguns fatores que podem influenciar nessa redução de tempo. É possível que a criança já tenha entendido como usar o comando de giro, pois precisou usá-lo anteriormente. Outra possibilidade é que, na fase de depuração os blocos já estão encaixados, e editar o algoritmo é mais fácil que construir um algoritmo sem nenhuma indicação inicial. A combinação desses

dois fatores, compreensão do giro e edição mais fácil, também pode ter influenciado no resultado.

6.3 ENTREVISTAS

As professoras responsáveis pelas turmas participantes foram entrevistadas após o término das atividades com as crianças. As questões aplicadas partiram de temas gerais, como a percepção do professor sobre o contato de crianças com tecnologias digitais (Q1), até a questão específica sobre a avaliação da RoPE AR (Q4). Aqui as respostas são apresentadas da temática mais específica para a mais geral.

Para a pergunta *Q4 - Que problemas você percebe na proposta? Quais aspectos poderiam melhorar?*, as professoras consideraram não haver problemas perceptíveis na interface, ao menos do ponto de vista pedagógico. Além disso, elas utilizaram adjetivos positivos para opinar sobre a RoPE AR:

Prof_1: Eu acho bem legal, porque é convidativo para eles assim. Isso poderia ser usado como se a gente colocasse um retroprojeto pra eles?

Prof_3: [...] eu achei muito legal assim, porque eles gostaram assim, eles ficaram virados naquele robzinho. “E como é que comeu a maçã” eu achei engraçado né ele olhou por baixo onde é que vai a maçã né. Mas eu acho que ainda é um pouco distante.

Prof_4: Isso é legal, isso é muito bom, gostei bastante. Eu achei que você ia colocar ali, como a gente tá acostumado né. Mas no chão foi muito legal...

A Prof_1 questiona se a aplicação poderia ser usada com o projetor na posição padrão, apontado para a parede. Isso indica incompreensão da proposta da interface, que é permitir a interação ativa das crianças para resolver problemas, ao invés de consumir conteúdo projetado na parede. A Prof_4 também comenta que está acostumada ao uso do projetor, mas considera “*legal*” o método de apontá-lo para o chão. Por fim, ela avalia a RoPE AR como um equipamento seguro:

Prof_4: Antes de tudo isso eu iria explicar pra eles que isso ali é um projetor, que teriam que cuidar pra não bater, porque também a gente trabalha com projetor na parede né, então quando a gente traz a gente diz que só a prof pode mexer, eles não podem mexer. [...] Eu não vejo como um perigo para eles. Isso não ia cair em cima de uma criança e machucar. No máximo iria quebrar, no máximo teria um prejuízo. A gente se preocupa mais se fosse uma coisa muito pesada e poderia machucar uma criança.

A Prof_3 avalia positivamente a RoPE AR, mas acha que é uma realidade “*distante*” devido ao custo de implementar um espaço exclusivo para equipamentos “*caros*” e à ausência de “*formação tecnológica*” para os professores.

Prof_3: Mas eu acho que ainda é um pouco distante. Talvez daqui mais algum tempo, quando isso ficar mais acessível. Porque eu sei que um laboratório, informática e tudo, digital, isso é um investimento alto né, então fica mais nas universidades. [...] Porque nós professores, nós não temos a formação tecnológica né.

Quanto às aplicações da RoPE AR (Q3), a Prof_3 sugere a construção de jogos de tabuleiro, e a exploração musical para discriminação auditiva. A proposta é emitir o som do instrumento quando o RoPE chegar sobre ele. No mesmo sentido, a professora também sugere emitir o som de letras.

Prof_3: Poderia até fazer jogos de tabuleiro. Dependendo onde você vai, tem que avançar na casa, retroceder, voltar no jogo... que é interessante também né pros menores, porque eles gostam bastante. Tem jogos mais simples, jogos mais complexos. Tipo um caça palavras ali, pras crianças que são alfabetizadas seria interessante. [...] Ou ele faz um som e aparece... vamos supor, se for um instrumento musical, se aparece um som daquilo ele pode estar identificando, por exemplo, deu um som de trovão, tem a figura do trovão e ele vai ter que associar. É um outro jogo no mesmo [estilo], mas com outra função, que eu posso associar. Porque quem tá fazendo associação de sons, né, discriminação auditiva, brincadeiras de discriminação auditiva, ele pode chegar em determinado local, fazer um som, e daquilo a criança vai ter que identificar qual é pra seguir adiante. Ou até uma letra. Ele chega até uma letra, ele faz o som.

Já a Prof_4 sugere utilizar o princípio da detecção de posição do RoPE para contação de histórias. A interface poderia projetar um lobo, o qual o RoPE deveria seguir.

Prof_4: É, tipo se fosse uma contação de história? Vai contando uma história e o robô vai demonstrando um cenário, vai chegando perto do cenário e vai fazendo um personagem da história. Eu pensaria nisso... Tipo um lobo, chegou no mato e o robô segue o lobo, algo assim, e vai chegando...

Para a Q2 - *Que importância você daria para a visibilidade no aprendizado das crianças?*, todas as professoras afirmaram ser necessário o contato com materiais concretos.

Prof_1: Ah, eu acho importantíssimo, porque eu sou visual, eu aprendo muito mais olhando e ouvindo, presencial. Tirar dúvidas né, na hora... [...] tem que mostrar, aprende muito mais.

Prof_3: Ah, sim. Pros menores, até a criança de sete, oito anos, é sempre concreto né, é sempre a partir daquele material, que ele possa pegar, visualizar, pra ele internalizar. Porque o abstrato assim, ficar explicando muito, eles não vão conseguir fazer a associação. Eles precisam associar que uma coisa é ligada a outra. Então eles têm que pegar, eles têm que sentir, eles têm que ver. Essa forma ali, eles conseguiram ter os comandos na hora.

A Prof_3 ainda comenta, a partir de uma observação do pesquisador, a respeito de toquinhos de madeira que estavam presentes na sala. Segundo ela, materiais não-estruturados são imprescindíveis nos espaços educacionais:

Prof_3: A gente está acostumado com plástico. Então plástico, madeira, são outras possibilidades de construção. Esses dias ele [uma criança] tava fazendo um prédio, disse que tava fazendo um prédio. Ele empilhou todos os toquinhos assim. Então é isso mesmo, é construir ou construir uma cerca. O bloco é essa tentativa. Ele não é um objeto muito estruturado. A criança tem que ter contato com objetos não estruturados. Então ele não tem forma, ele não deve para um devido fim. Ele vai montar uma estrutura a partir daquele objeto que não tem formato. Ele que vai ter que imaginar e dar forma àquele objeto. Então o objeto não estruturado é imprescindível que tenha no espaço de educação.

Os blocos de código, assim como os toquinhos de madeira, também são elementos não estruturados. A criança os organiza para definir uma rota, uma história, ou um comportamento para o brinquedo RoPE. Neste sentido, os materiais do ambiente visitado e a proposta da pesquisa se assemelham: são objetos concretos e não estruturados, que a criança estrutura conforme seu desejo.

Por fim, a questão mais geral (Q1): *O que você pensa sobre o contato de crianças com tecnologias digitais?* teve as respostas a seguir:

Prof_1: [...] é uma forma educativa de inserir eles na informatização, porque em casa o celular e a tecnologia é só voltada pro YouTube, coisas que não são tão úteis. E se a gente puder introduzir isso de uma forma que eles procurem no Google o conhecimento, informática... isso é bem legal. Interessante. Mudar o foco do YouTube...

Prof_3: Eu acho fantástico, é uma pena que nós não temos né. Na rede pública é mais difícil porque ter mais equipamentos né, de pesquisa com as crianças, como uma mesa digital, isso é muito custoso né, pro município ter.

Prof_4: Ah, eu acho necessário. Eu acho necessário, mas também tem que ter o equilíbrio assim. Tu não pode deixar a criança só com tecnologia, vamos supor, o celular, que é o que eles tem em mãos, né, que a gente viu que foi o que aconteceu com alguns, que passaram a pandemia toda com o celular na mão, tem que ter o equilíbrio.

As respostas da Q1 evidenciam a percepção negativa sobre o acesso exagerado, não supervisionado e passivo das crianças a equipamentos eletrônicos para assistir vídeos na internet. A Prof_1 cita a necessidade de “*mudar o foco do YouTube*”, e a Prof_4 fala que não se “*pode deixar a criança ‘só’ com a tecnologia*”.

As respostas evidenciam uma percepção positiva da RoPE AR. As sugestões de aplicações da interface incluíram jogos de tabuleiro, atividades para reconhecer letras e sons, e também contação de histórias. Uma conclusão é que as professoras perceberam potencial pedagógico no uso do princípio de aplicar realidade aumentada associada a objetos concretos, à captação de posições desses objetos, e à emissão de sons. O suporte a essas aplicações, entretanto, exigiria uma plataforma de configuração, como o exemplo da Ct Puzzle Platform, mas que foge do foco da atual pesquisa.

As professoras não tiveram contato com a etapa de instalação e conexão dos equipamentos. Essa etapa apresentou dificuldades de conexão em uma das salas, conforme relatado na descrição do contexto de pesquisa. É possível que apresentassem uma opinião menos positiva a respeito da interface em caso de a usarem por conta própria.

7 CONCLUSÃO

O tema central desta pesquisa foi a depuração de algoritmos por crianças. A motivação para tal temática é a presença da depuração como um dos princípios de design do ambiente Logo. Esse ambiente, desenvolvido por Papert e sua equipe na década de 1980, foi uma das primeiras ferramentas na área da Informática na Educação com foco em possibilitar a crianças aprenderem programação. E mais do que aprender programação, a proposta é permitir que a criança aprenda sobre a sua própria forma de pensar.

O ambiente Logo sofreu alterações de design desde a sua criação. A tartaruga robótica originou brinquedos programáveis, os quais possuem diversos formatos que vão desde outros animais, veículos, até personagens robóticos de madeira. Deste modo, a aparência lúdica desses brinquedos os tornou mais aderentes ao público infantil. Essa pesquisa apresentou 56 exemplos desses brinquedos, através de um mapeamento industrial sistemático.

Esse mapeamento também apresenta outra transformação de design: novos métodos de inserir comandos para esses brinquedos/computadores executarem. O que antes as crianças faziam usando teclados, exigindo alfabetização, agora é possível por blocos de madeira, blocos eletrônicos, aplicativos de smartphone, botões acoplados, e botões de controle remoto. Essa pesquisa classificou essas interfaces em dois grupos: tangíveis e virtuais.

Argumentamos que as interfaces tangíveis, de modo isolado, não oferecem um suporte ideal para que crianças depurem algoritmos. Blocos de madeira carecem de mecanismos que promovam a visibilidade, descrita por Norman (1988), e indiquem qual ação está sendo executada em cada momento. As alternativas existentes, como o Cubetto e o Robo-Blocks necessitam de componentes eletrônicos específicos. Essa dependência não é um problema em produtos acabados, mas o processo de design de ferramentas educacionais necessita de diversas iterações e alterações, nesse caso depender de componentes eletrônicos pode atrasar o desenvolvimento. Outro aspecto, ligado à visibilidade, é seu impacto na construção do modelo mental, ou seja, o entendimento que o usuário cria sobre como um dispositivo funciona.

As interfaces virtuais, por outro lado, também não são adequadas para crianças, principalmente quando utilizadas de modo passivo e sem supervisão (STIGLIC; VINER, 2019). Esses equipamentos também são geralmente de uso individual, o que não favorece a colaboração e interações interpessoais,

fundamentais para o aprendizado.

Esse trabalho propôs a RoPE AR como uma alternativa. Ela utiliza realidade aumentada projetiva para adicionar elementos virtuais sobre elementos tangíveis. Deste modo, a criança pode criar algoritmos com blocos de papelão e os elementos virtuais destacam e aumentam o significado desses blocos. O uso de papelão e elementos gráficos facilita criar de novos designs de blocos e atividades, pois é um material acessível.

Partindo desta proposta, a presente pesquisa analisou a existência de trabalhos similares. Com isso, apresentou trabalhos que aplicam RA projetiva na medicina, na indústria e também em produtos educacionais. O trabalho de Beraza, Pina e Demo (2016) se destaca por também aplicar realidade aumentada projetiva em um tapete virtual para Bee-Bot, a qual é programada por botões. O diferencial da RoPE AR é explorar a construção de algoritmos com blocos tangíveis, e promover a interação dos blocos e do brinquedo programável com objetos virtuais.

Neste contexto, surgiu a seguinte questão de pesquisa: como crianças que são público-alvo de um brinquedo programável, poderiam interagir com uma interface de programação focada em permitir visualizar os passos e a execução de um algoritmo? Quatro objetivos específicos operacionalizam a resposta a essa questão.

O primeiro objetivo - Descrever sistemas e aplicações que utilizam realidade aumentada projetiva - foi atendido pela fundamentação teórica. O trabalho de Wrzesien et al. (2015), destinado ao tratamento de fobias, e o trabalho de Roberto (2013), uma plataforma de realidade aumentada, são exemplos citados. Esses trabalhos evidenciaram a viabilidade da técnica escolhida, e a sua aplicabilidade a ferramentas educacionais.

O segundo objetivo - Mapear as principais categorias interfaces de brinquedos programáveis existentes - foi atendido através de um mapeamento industrial sistemático. Esse mapeamento listou os brinquedos disponíveis comercialmente, e também incluiu os brinquedos citados em revisões como as de Hamilton et al. (2020) e de Yu e Roque (2019). Esse mapeamento evidenciou que a realidade aumentada é pouco usada em brinquedos programáveis.

O terceiro objetivo — Construir a interface de realidade aumentada projetiva para o brinquedo RoPE — ocorreu com a modelagem e implementação da RoPE AR, descritas no Capítulo 4. Essa construção abrangeu alterar o firmware do RoPE para se comunicar com um dispositivo móvel; implementar essa comunicação em tal ambiente; desenhar os blocos tangíveis; implementar a projeção

sobre os mesmos. Nessa etapa também foi implementada a integração com a Ct Puzzle Platform, que viabilizou configurar os ambientes projetados pela RoPE AR e também armazenar as interações em registros para posterior processamento.

Por fim, a interface possibilitou atender ao objetivo quarto - Experimentar a interface em atividades com crianças e coletar eventos de interação. Essa experimentação ocorreu em um CDI, com a participação de 20 crianças. Elas programaram e corrigiram algoritmos, em duplas e trios, com auxílio do pesquisador e também de professoras. Vídeos das atividades e entrevistas com as professoras formam os dados captados nesta etapa.

Após o cumprimento destes objetivos, por fim, o Capítulo 6 tenta responder à questão de pesquisa, de como as crianças interagiram com tal ambiente. Para isso, analisa vídeos das interações e entrevistas com as professoras de modo indutivo e dedutivo. As conclusões, não definitivas, são as seguintes:

1. As crianças entendem o significado dos desenhos de cada bloco (Quadro 18), e tendem a falar de “esquerda” e “direita” como “pro lado”
2. As crianças relacionam as cores dos blocos com as cores dos botões do brinquedo RoPE. Também entendem como encaixar esses blocos para formar sequências válidas, com o bloco verde na primeira posição (Figura 47a)
3. As crianças percebem quando elementos projetados surgem e somem do mapa
4. As crianças estranharam o fato de um elemento virtual desaparecer, e procuram a maçã dentro da “barriga” do robô. Há forte integração entre virtualidade e realidade, o que e as crianças podem não distinguir (Quadro 19)
5. As crianças reagem à função de destacar os blocos do algoritmo encaixando ainda mais blocos na sequência (Figura 48b), mas isso aconteceu apenas em três momentos
6. O engajamento na tarefa é indicado por (Figura 49 e Figura 49) menos de 3% das ações representaram estados de distração ou repouso.
7. O efeito sonoro e visual do brinquedo capturando a maçã projetada parece influenciar no engajamento
8. As crianças não perceberam ou não valorizaram o destaque dos blocos em cada movimento do brinquedo. Sua atenção se voltou ao brinquedo

9. As crianças não depuram os algoritmos por conta própria. Entretanto, as atividades focadas em depuração permitiram avaliar o algoritmo, identificar erros, localizar e corrigir bugs. Mas isso dependeu da orientação do pesquisador, e em apenas uma atividade permite afirmar que a realidade aumentada influencia a depuração, ao menos na faixa etária estudada
10. As crianças têm forte atração pelo feedback imediato (NORMAN, 1998) fornecido pelo clique dos botões, o que as fez esquecerem de usar os blocos de papelão
11. As crianças precisam de mais tempo nas fases de programação envolvendo giro (Figura 51)
12. As fases de depuração demandam menos tempo de resolução (Figura 51) do que as tarefas de programação, mas não foi possível afirmar qual o motivo
13. As professoras avaliam positivamente a aplicação proposta, e sugerem atividades condizentes com a união de interfaces tangíveis e realidade aumentada. Também não temem que as crianças danifiquem os blocos de papelão
14. A similaridade de cores entre blocos é um fator causador de erros

7.1 CONTRIBUIÇÕES DA DISSERTAÇÃO

Essa pesquisa teve abordagem exploratória, e sua principal contribuição é apresentar resultados empíricos sobre uma temática que carece de ser explorada. Neste sentido, as conclusões podem servir de base para próximos trabalhos de caráter quantitativo para avaliar hipóteses aqui levantadas.

A construção e avaliação da RoPE AR é também uma contribuição. O fato de estar conectada a uma plataforma de configuração de testes permite definir outros mapas, sem a necessidade de recompilar o software. Deste modo o seu uso não fica restrito aos objetivos do presente trabalho.

O mapeamento de 56 brinquedos programáveis em uma planilha pública¹ também é um subproduto útil para pesquisas futuras. O maior número de brinquedos analisados em pesquisas (HAMILTON et al., 2020; YU; ROQUE, 2019) foi 30, o que indica que os resultados obtidos na presente pesquisa são mais abrangentes. Além dos 56 brinquedos, a lista contém brinquedos encontrados que não foram analisados, e podem integrar o corpus mapeado. O fato da planilha ser pública e manter o histórico de alterações permite adição de novos registros sem risco de perder a informação atual.

¹ <<https://bit.ly/35cUbZ1>>

A implementação da biblioteca de comunicação por Bluetooth com o RoPE via sistema Android² também é um subproduto que pode apoiar aplicações que necessitem comunicar ações ou obter informações do brinquedo a curta distância.

Por fim, as entrevistas transcritas representam uma fonte de análise para futuras pesquisas. Este trabalho analisou apenas os aspectos focados na atividade, mas o questionário aberto apresenta outras observações das professoras. Um tema abordado foi a formação tecnológica das professoras de educação infantil.

7.2 TRABALHOS FUTUROS

Foram vislumbradas algumas possibilidades de investigação futuras a esta pesquisa. Entre elas, está a construção de uma plataforma onde as professoras possam definir mapas, elementos virtuais, e sons presentes nas atividades. Seria um modo de suportar as aplicações sugeridas, como contação de histórias e jogos, e associar a criatividade das professoras ao ambiente proposto.

Ao externalizar os comandos que podem ser programados no brinquedo RoPE, não há mais a limitação de espaço físico nos botões do brinquedo. Pode-se criar blocos com finalidades específicas, como programar sons, movimentos de amplitudes variadas e programar efeitos luminosos. Também podem ser criados blocos que quando executados, causem efeitos não apenas no brinquedo, mas também no mapa, como mudar a cor do tapete ou mostrar outro cenário. Uma linha de pesquisa seria, portanto, identificar como usar esse potencial para favorecer o aprendizado em diferentes faixas etárias.

A identificação da posição do brinquedo RoPE também abre espaço para novas aplicações. Uma aplicação possível seria criar um bloco de caneta, semelhante à presente no ambiente Logo. Contudo, diferente de usar uma caneta física, a marcação da trajetória do brinquedo seria uma linha projetada. Essa linha pode ter, portanto, diferentes cores e formas.

Outra possibilidade é avaliar métodos alternativos de comunicar algoritmos, como a fala ou gestos. Tendo em vista que é possível programar o RoPE por Bluetooth, um adulto escondido pode observar a criança enquanto ela fala ou gesticula e programar o brinquedo de acordo. Isso poderia gerar *insights* de formas mais naturais da criança programar.

² <https://github.com/cesarviana/rope_ar>

7.3 ASPECTOS ÉTICOS

Essa pesquisa foi aprovada pelo Comitê de Ética em Pesquisa da Univali, conforme parecer 4.027.404. Nenhuma criança ou professora foi identificada, seja por informações pessoais ou imagens. As professoras autorizaram o compartilhamento das entrevistas, e o acesso ao CDI foi autorizado pela direção e pela Secretaria Municipal de Educação.

REFERÊNCIAS

- ANZOATEGUI, L. G. C.; PEREIRA, M. I. A. R.; JARRIN, M. d. C. S. Cubetto for preschoolers: Computer programming code to code. In: **2017 International Symposium on Computers in Education (SIIE)**. Lisbon: IEEE, 2017. p. 1–5. ISBN 978-1-5386-0648-3. Disponível em: <<https://ieeexplore.ieee.org/document/8259649/>>.
- AZUMA, R.; BAILLOT, Y.; BEHRINGER, R.; FEINER, S.; JULIER, S.; MACINTYRE, B. Recent advances in augmented reality. **IEEE Computer Graphics and Applications**, v. 21, n. 6, p. 34–47, 2001. Disponível em: <<http://ieeexplore.ieee.org/document/963459/>>.
- BARR, V.; STEPHENSON, C. Bringing Computational Thinking to K-12: What is Involved and What is the Role of the Computer Science Education Community? **ACM Inroads**, v. 2, n. 1, p. 48–54, 2011.
- BBC Learning. **What is computational thinking? - Introduction to computational thinking - KS3 Computer Science Revision**. 2015. Library Catalog: www.bbc.co.uk. Disponível em: <<https://www.bbc.co.uk/bitesize/guides/zp92mp3/revision/1>>.
- BERAZA, I.; PINA, A.; DEMO, B. Soft & hard ideas to improve interaction with robots for kids & teachers. In: . [S.l.: s.n.], 2010. p. 549–555.
- BERS, M. U. **Blocks to robots: learning with technology in the early childhood classroom**. New York: Teachers College Press, 2008. OCLC: ocn156994433. ISBN 978-0-8077-4848-0 978-0-8077-4847-3.
- _____. Coding, playgrounds and literacy in early childhood education: The development of KIBO robotics and ScratchJr. In: **2018 IEEE Global Engineering Education Conference (EDUCON)**. Tenerife: IEEE, 2018. p. 2094–2102. ISBN 978-1-5386-2957-4. Disponível em: <<https://ieeexplore.ieee.org/document/8363498/>>.
- _____. Coding as another language: a pedagogical approach for teaching computer science in early childhood. **Journal of Computers in Education**, v. 6, n. 4, p. 499–528, dez. 2019. ISSN 2197-9987, 2197-9995. Disponível em: <<http://link.springer.com/10.1007/s40692-019-00147-3>>.
- BERS, M. U.; FLANNERY, L.; KAZAKOFF, E. R.; SULLIVAN, A. Computational thinking and tinkering: Exploration of an early childhood robotics curriculum. **Computers & Education**, v. 72, p. 145–157, mar. 2014. ISSN 03601315. Disponível em: <<https://linkinghub.elsevier.com/retrieve/pii/S0360131513003059>>.
- BEŞEVLI, C.; SALMAN, E.; GOKSUN, T.; UREY, H.; ÖZCAN, O. MaR-t: Designing a projection-based mixed reality system for nonsymbolic math development of preschoolers: Guided by theories of cognition and learning. In: **Proceedings of the 18th ACM International Conference on Interaction Design and Children**. ACM, 2019. p. 280–292. ISBN 978-1-4503-6690-8. Disponível em: <<https://dl.acm.org/doi/10.1145/3311927.3323147>>.
- BIMBER, O.; RASKAR, R. **Spatial Augmented Reality: Merging Real and Virtual Worlds**. Wellesley: CRC Press, 2005. Google-Books-ID: JEu3BgAAQBAJ. ISBN 978-1-4398-6494-4.
- BLIKSTEIN, P. Analyzing engineering design through the lens of computation. v. 1, n. 2, p. 151–186, 2014. ISSN 1929-7750. Number: 2. Disponível em: <<https://learning-analytics.info/index.php/JLA/article/view/3541>>.
- BONI, V.; QUARESMA, S. J. Aprendendo a entrevistar: como fazer entrevistas em ciências sociais. v. 1, p. 13, 2005. Disponível em: <<https://periodicos.ufsc.br/index.php/emtese/article/view/18027/16976>>.

- BORDINI, A.; AVILA, C. M. O.; WEISSHAHN, Y.; CUNHA, M. M. d.; CAVALHEIRO, S. A. d. C.; FOSS, L.; AGUIAR, M. S.; REISER, R. H. S. Computação na Educação Básica no Brasil: o Estado da Arte. **Revista de Informática Teórica e Aplicada**, v. 23, n. 2, p. 210, dez. 2016. ISSN 21752745, 01034308. Disponível em: <<http://www.seer.ufrgs.br/index.php/rita/article/view/RITA-VOL23-NR2-210>>.
- BORNEMANN, K.; LAWONN, K.; HEINRICH, F.; HANSEN, C. Exploration of medical volume data in projective augmented reality: An interactive demonstration. p. 3, 2020.
- BRACKMANN, C. P. **Desenvolvimento do Pensamento Computacional Através de Atividades Desplugadas na Educação Básica**. Tese (Tese) — Universidade Federal do Rio Grande do Sul, Porto Alegre, BR-RS, 2017.
- BRASIL. Lei nº 8069, de 13 de julho de 1990. Dispõe sobre o Estatuto da Criança e do Adolescente e dá outras providências. **1990**, 1990. Brasília. Disponível em: <http://www.planalto.gov.br/ccivil_03/leis/l8069.htm>.
- BRENNAN, K.; RESNICK, M. New frameworks for studying and assessing the development of computational thinking. p. 25, 2012.
- BURLESON, W. S.; HARLOW, D. B.; NILSEN, K. J.; PERLIN, K.; FREED, N.; JENSEN, C. N.; LAHEY, B.; LU, P.; MULDER, K. Active Learning Environments with Robotic Tangibles: Children's Physical and Virtual Spatial Programming Experiences. **IEEE Transactions on Learning Technologies**, v. 11, n. 1, p. 96–106, jan. 2018. ISSN 1939-1382, 2372-0050. Disponível em: <<https://ieeexplore.ieee.org/document/7971992/>>.
- CARVER, S.; RISINGER, S. **Improving children's debugging skills**. [S.l.: s.n.], 1987. Pages: 171.
- CARVER, S. M.; KLAHR, D. Assessing Children's Logo Debugging Skills with a Formal Model. **Journal of Educational Computing Research**, v. 2, n. 4, p. 487–525, nov. 1986. ISSN 0735-6331, 1541-4140. Disponível em: <<http://journals.sagepub.com/doi/10.2190/KRD4-YNHH-X283-3P5V>>.
- CATLIN, D.; KANDLHOFER, M.; HOLMQUIST, S.; CSIZMADIA, A. P.; CSIZMADIA, A. P.; ANGEL-FERNANDEZ, J.; CABIBIHAN, J.-J. EduRobot Taxonomy and Papert's Paradigm. In: **Constructionism 2018**. Vilnius, Lithuania: [s.n.], 2018. p. 11. Disponível em: <https://www.researchgate.net/publication/327552838_EduRobot%27s_Taxonomy_and_Papert%27s_Paradigm>.
- COOPER, S.; DANN, W.; PAUSCH, R. **Alice: A 3-D Tool for Introductory Programming Concepts**. Evansville, IN, USA: Consortium for Computing Sciences in Colleges, 2000. 107–116 p. ISSN 1937-4771.
- DAVIDSON, S.; MURPHY, C. **1.2 Decomposition - Pillars of Computational Thinking**. 2018. Library Catalog: www.coursera.org. Disponível em: <<https://www.coursera.org/lecture/computational-thinking-problem-solving/1-2-decomposition-phmNk>>.
- DUBROFSKY, E. **Homography Estimation**. Tese (Dissertação de Mestrado) — University of British Columbia, Vancouver, 2009. Disponível em: <https://www.cs.ubc.ca/grads/resources/thesis/May09/Dubrofsky_Elan.pdf>.
- FALCÃO, T. P. d. R. **Design de interfaces tangíveis para aprendizagem de conceitos matemáticos no Ensino Fundamental**. Tese (Dissertação de Mestrado) — Universidade Federal do Pernambuco, Recife, fev. 2007.
- FIALA, M. Designing Highly Reliable Fiducial Markers. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 32, n. 7, p. 1317–1324, jul. 2010. ISSN 1939-3539. Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence.
- FISHKIN, K. A taxonomy for and analysis of tangible interfaces. **Personal and Ubiquitous Computing**, v. 8, n. 5, set. 2004. ISSN 1617-4909, 1617-4917. Disponível em: <<http://link.springer.com/10.1007/s00779-004-0297-4>>.

FLANNERY, L. P.; SILVERMAN, B.; KAZAKOFF, E. R.; BERS, M. U.; BONTÁ, P.; RESNICK, M. Designing ScratchJr: support for early childhood learning through computer programming. In: **Proceedings of the 12th International Conference on Interaction Design and Children - IDC '13**. New York, New York: ACM Press, 2013. p. 1–10. ISBN 978-1-4503-1918-8. Disponível em: <<http://dl.acm.org/citation.cfm?doid=2485760.2485785>>.

GAMMA, E.; HELM, R.; JOHNSON, R.; VLISSIDES, J. **Design Patterns: Elements of Reusable Object-Oriented Software**. Illustrated edition. Addison-Wesley, 1994. ISBN 978-0-201-63361-0 0-201-63361-2. Disponível em: <<http://gen.lib.rus.ec/book/index.php?md5=4963a7703acf363425e8099d6b12d2dc>>.

GASPAR. **Proposta pedagógica da rede municipal: educação infantil**. 2010. Organizadoras: Julice Dias, Patrícia Helena dos Santos e Sanira Cristina Dias. Disponível em: <https://static.fecam.net.br/uploads/878/arquivos/63635_Proposta_Pedagogica_Educacao_Infantil_Paginas.pdf>.

GIL, A. C. **Métodos e técnicas de pesquisa social**. 6. ed. São Paulo: Atlas, 2008. ISBN 978-85-224-5142-5.

GRAY, C.; MacBlain, S. **Learning Theories in Childhood**. [S.l.]: SAGE, 2015. Google-Books-ID: 6vuICwAAQBAJ. ISBN 978-1-4739-3380-4.

HAMILTON, M.; CLARKE-MIDURA, J.; SHUMWAY, J. F.; LEE, V. R. An Emerging Technology Report on Computational Toys in Early Childhood. **Technology, Knowledge and Learning**, v. 25, n. 1, p. 213–224, mar. 2020. ISSN 2211-1662, 2211-1670. Disponível em: <<http://link.springer.com/10.1007/s10758-019-09423-8>>.

HEIKKILÄ, M.; MANNILA, L. Debugging in programming as a multimodal practice in early childhood education settings. v. 2, p. 42, 2018.

HELJAKKA, K.; IHAMAKI, P.; TUOMI, P.; SAARIKOSKI, P. Gamified Coding: Toy Robots and Playful Learning in Early Education. In: **2019 International Conference on Computational Science and Computational Intelligence (CSCI)**. Las Vegas, NV, USA: IEEE, 2019. p. 800–805. ISBN 978-1-72815-584-5. Disponível em: <<https://ieeexplore.ieee.org/document/9071010/>>.

HOED, R. M. Análise da evasão em cursos superiores : o caso da evasão em cursos superiores da área de computação. dez. 2016. Accepted: 2017-02-15T15:45:28Z. Disponível em: <<https://repositorio.unb.br/handle/10482/22575>>.

HORN, M.; BERS, M. Tangible Computing. In: FINCHER, S. A.; ROBINS, A. V. (Ed.). **The Cambridge Handbook of Computing Education Research**. 1. ed. Cambridge University Press, 2019. p. 663–678. ISBN 978-1-108-65455-5 978-1-108-49673-5 978-1-108-72189-9. Disponível em: <https://www.cambridge.org/core/product/identifier/9781108654555%23CN-bp-22/type/book_part>.

HORN, M. S.; CROUSER, R. J.; BERS, M. U. Tangible interaction and learning: the case for a hybrid approach. **Personal and Ubiquitous Computing**, v. 16, n. 4, p. 379–389, abr. 2012. ISSN 1617-4909, 1617-4917. Disponível em: <<http://link.springer.com/10.1007/s00779-011-0404-2>>.

HOURCADE, J. P. **Child-Computer Interaction**. [s.n.], 2015. Disponível em: <<http://homepage.cs.uiowa.edu/~hourcade/book/index.php>>.

HU, F.; ZEKELMAN, A.; HORN, M.; JUDD, F. Strawbies: explorations in tangible programming. In: **Proceedings of the 14th International Conference on Interaction Design and Children - IDC '15**. Boston, Massachusetts: ACM Press, 2015. p. 410–413. ISBN 978-1-4503-3590-4. Disponível em: <<http://dl.acm.org/citation.cfm?doid=2771839.2771866>>.

ILIC, U.; HASESKI, H. I.; TUGTEKIN, U. Publication Trends Over 10 Years of Computational Thinking Research. **Contemporary Educational Technology**, p. 131–153, abr. 2018. ISSN 1309-517X. Disponível em: <<http://dergipark.gov.tr/doi/10.30935/cet.414798>>.

JACK, K. **Video demystified: a handbook for the digital engineer**. 3rd ed. ed. [S.l.]: LLH Technology Pub, 2001. (Demystifying technology series). ISBN 978-1-878707-56-7.

K-12 Computer Science Framework. **K-12 Computer Science Framework**. 2016. Disponível em: <<http://www.k12cs.org>>.

KITCHENHAM, B.; CHARTERS, S. **Guidelines for performing Systematic Literature Reviews in Software Engineering**. [S.l.: s.n.], 2007.

LAVIOLE, J.; THEVIN, L.; ALBOUYS-PERROIS, J.; BROCK, A. Nectar: Multi-user spatial augmented reality for everyone: Three live demonstrations of educative applications. In: **Proceedings of the Virtual Reality International Conference - Laval Virtual**. Association for Computing Machinery, 2018. (VRIC '18), p. 1–6. ISBN 978-1-4503-5381-6. Disponível em: <<https://doi.org/10.1145/3234253.3234317>>.

LIETO, M. C. D.; INGUAGGIATO, E.; CASTRO, E.; CECCHI, F.; CIONI, G.; DELL'OMO, M.; LASCHI, C.; PECINI, C.; SANTERINI, G.; SGANDURRA, G.; DARIO, P. Educational Robotics intervention on Executive Functions in preschool children: A pilot study. **Computers in Human Behavior**, v. 71, p. 16–23, jun. 2017. ISSN 07475632. Disponível em: <<https://linkinghub.elsevier.com/retrieve/pii/S0747563217300195>>.

LIU, Z.; ZHI, R.; HICKS, A.; BARNES, B. Understanding problem solving behavior of 6–8 graders in a debugging game. p. 30, 2017.

LYE, S. Y.; KOH, J. H. L. Review on teaching and learning of computational thinking through programming: What is next for K-12? **Computers in Human Behavior**, v. 41, p. 51–61, dez. 2014. ISSN 07475632. Disponível em: <<https://linkinghub.elsevier.com/retrieve/pii/S0747563214004634>>.

MARKOPOULOS, P.; READ, J.; MACFARLANE, S.; HöYSNIEMI, J. Evaluating Children's Interactive Products. **Evaluating Children's Interactive Products**, jan. 2008.

MARTINS, R. R. **Desenvolvimento de uma metodologia para explorar habilidades de engenharia através do uso de brinquedos programáveis**. Tese (Trabalho Técnico-científico de Conclusão de Curso) — Universidade do Vale do Itajaí, Itajaí, 2016.

MCCAULEY, R.; FITZGERALD, S.; LEWANDOWSKI, G.; SIMON, B.; THOMAS, L.; ZANDER, C. Debugging: a review of the literature from an educational perspective. v. 18, n. 2, p. 28, 2008.

MCCRACKEN, M.; ALMSTRUM, V.; DIAZ, D.; GUZDIAL, M.; HAGAN, D.; KOLIKANT, Y. B.-D.; LAXER, C.; THOMAS, L.; UTTING, I.; WILUSZ, T. A multi-national, multi-institutional study of assessment of programming skills of first-year CS students. p. 16, 2001.

MCNERNEY, T. From turtles to Tangible Programming Bricks: explorations in physical language design. **Personal and Ubiquitous Computing**, v. 8, n. 5, set. 2004. ISSN 1617-4909, 1617-4917. Disponível em: <<http://link.springer.com/10.1007/s00779-004-0295-6>>.

MEKNI, M.; LEMIEUX, A. Augmented reality: Applications, challenges and future trends. p. 10, 2014.

METZGER, J. P. **Desenvolvimento de uma interface de programação tangível microcontrolada para o brinquedo RoPE que estimule a contação de histórias**. Tese (Trabalho Técnico-científico de Conclusão de Curso) — Universidade do Vale do Itajaí, Itajaí, 2018.

MILGRAM, P.; TAKEMURA, H.; UTSUMI, A.; KISHINO, F. Augmented reality: a class of displays on the reality-virtuality continuum. In: DAS, H. (Ed.). Boston, MA: [s.n.], 1994. p. 282–292. Disponível em: <<http://proceedings.spiedigitallibrary.org/proceeding.aspx?articleid=981543>>.

MONTESSORI, M. **O Segredo da Infância**. 1ª edição. ed. [S.l.]: Kíron, 2019. ISBN 978-85-940902-2-5.

- NAM, K. W.; KIM, H. J.; LEE, S. Connecting Plans to Action: The Effects of a Card-Coded Robotics Curriculum and Activities on Korean Kindergartners. **The Asia-Pacific Education Researcher**, v. 28, n. 5, p. 387–397, out. 2019. ISSN 0119-5646, 2243-7908. Disponível em: <<http://link.springer.com/10.1007/s40299-019-00438-4>>.
- NORMAN, D. **The Design of Everyday Things**. Nova York: Doubleday, 1990. ISBN 9780385267748. Disponível em: <<https://books.google.com.br/books?id=b09jQgAACAAJ>>.
- NOSCHANG, L. F.; PELZ, F.; JESUS, E. de; RAABE, A. Portugal studio: Uma ide para iniciantes em programação. In: **Anais do XXII Workshop sobre Educação em Computação**. Porto Alegre, RS, Brasil: SBC, 2014. p. 1–10. ISSN 2595-6175. Disponível em: <<https://sol.sbc.org.br/index.php/wei/article/view/10954>>.
- PAPERT, S. Teaching Children Thinking. **Programmed Learning and Educational Technology**, v. 9, n. 5, p. 245–255, set. 1972. ISSN 0033-0396. Disponível em: <<https://www.tandfonline.com/doi/full/10.1080/1355800720090503>>.
- _____. **Mindstorms: children, computers, and powerful ideas**. New York: Basic Books, 1980. ISBN 978-0-465-04627-0.
- _____. An exploration in the space of mathematics educations. **International Journal of Computers for Mathematical Learning**, v. 1, n. 1, 1996. ISSN 1382-3892, 1573-1766. Disponível em: <<http://link.springer.com/10.1007/BF00191473>>.
- PAPERT, S.; HAREL, I. Situating Constructionism. p. 17, 1991.
- PARVEAU, M.; ADDA, M. 3iVClass: a new classification method for Virtual, Augmented and Mixed Realities. **Procedia Computer Science**, v. 141, p. 263–270, 2018. ISSN 18770509. Disponível em: <<https://linkinghub.elsevier.com/retrieve/pii/S1877050918318301>>.
- PIAGET, J. Development and Learning. **Journal of Research In Science Teaching**, v. 2, p. 176–186, 1964.
- PILA, S.; ALADÉ, F.; SHEEHAN, K. J.; LAURICELLA, A. R.; WARTELLA, E. A. Learning to code via tablet applications: An evaluation of Daisy the Dinosaur and Kodable as learning tools for young children. **Computers & Education**, v. 128, p. 52–62, jan. 2019. ISSN 03601315. Disponível em: <<https://linkinghub.elsevier.com/retrieve/pii/S0360131518302422>>.
- PINHEIRO, L. A. **Análise da participação de crianças em atividades com brinquedos programáveis**. Tese (Doutorado) — Universidade do Vale do Itajaí, Itajaí, 2016.
- PINKAEW, C.; WARANUSAST, R. Interactive tangible word game. In: **2014 Third ICT International Student Project Conference (ICT-ISPC)**. IEEE, 2014. p. 51–54. ISBN 978-1-4799-5573-2 978-1-4799-5572-5 978-1-4799-5571-8. Disponível em: <<http://ieeexplore.ieee.org/document/6923216/>>.
- PLOWMAN, L.; LUCKIN, R. Interactivity, interfaces, and smart toys. **Computer**, v. 37, n. 2, p. 98–100, fev. 2004. ISSN 0018-9162. Disponível em: <<http://ieeexplore.ieee.org/document/1266302/>>.
- PUGNALI, A.; SULLIVAN, A.; BERS, M. U. The Impact of User Interface on Young Children's Computational Thinking. **Journal of Information Technology Education: Innovations in Practice**, v. 16, p. 171–193, 2017. ISSN 2165-3151, 2165-316X. Disponível em: <<https://www.informingscience.org/Publications/3768>>.
- RAABE, A.; RODRIGUES, A. J.; SANTANA, A. M.; VIEIRA, M. V.; ROSÁRIO, T.; CARNEIRO, A. C. R. Brinquedos de Programar na Educação Infantil: Um estudo de Caso. In: . Maceió, Alagoas, Brasil: [s.n.], 2015. p. 42. Disponível em: <<http://br-ie.org/pub/index.php/wie/article/view/4985>>.

RAABE, A.; ROSÁRIO, T.; MARTINS, R.; SANTANA, A. L. M.; SOUZA, F. T.; SILVA, R. RoPE - Brinquedo de Programar e Plataforma de Aprender. In: . Recife, Pernambuco, Brasil: [s.n.], 2017. p. 1119. Disponível em: <<http://www.br-ie.org/pub/index.php/wie/article/view/7349>>.

REPISO, A. G. V. M.; GONZÁLEZ, Y. A. C. Robotics to develop computational thinking in early Childhood Education. **Comunicar**, v. 27, n. 59, p. 63–72, abr. 2019. ISSN 1134-3478, 1988-3293. Disponível em: <<https://www.revistacomunicar.com/index.php?contenido=detalles&numero=59&articulo=59-2019-06>>.

RESCH, C. **Enhancing Projective Spatial Augmented Reality in Industry: A Model Based Approach for Registration and Calibration**. Dissertation, 2016.

ROBERTO, R. A.; FREITAS, D. Q. d.; SIMOES, F. P. M.; TEICHRIEB, V. A Dynamic Blocks Platform Based on Projective Augmented Reality and Tangible Interfaces for Educational Activities. In: **2013 XV Symposium on Virtual and Augmented Reality**. Cuiabá - Mato Grosso, Brazil: IEEE, 2013. p. 1–9. ISBN 978-0-7695-5001-5. Disponível em: <<http://ieeexplore.ieee.org/document/6655756/>>.

ROGERS, Y.; SHARP, E.; PREECE, J. **Design de Interação: Além da interação humano-computador**. 3. ed. Porto Alegre, BR-RS: Bookman, 2013. ISBN 978-85-8260-006-1.

SAMS, J. Blog, **Introducing Renderscript**. 2011. Android Developers Blog. Disponível em: <<https://android-developers.googleblog.com/2011/02/introducing-renderscript.html>>.

SAND, O.; BÜTTNER, S.; PAELKE, V.; RÖCKER, C. smARt.assembly – projection-based augmented reality for supporting assembly workers. In: LACKEY, S.; SHUMAKER, R. (Ed.). **Virtual, Augmented and Mixed Reality**. Cham: Springer International Publishing, 2016. (Lecture Notes in Computer Science), p. 643–652. ISBN 978-3-319-39907-2.

SANTANA, A. **Análise do processo metodológico de montagem de um brinquedo de programar**. Dissertação de Mestrado, 2015.

SAPOUNIDIS, T.; DEMETRIADIS, S.; PAPADOPOULOS, P. M.; STAMOVLASIS, D. Tangible and graphical programming with experienced children: A mixed methods analysis. **International Journal of Child-Computer Interaction**, v. 19, p. 67–78, 2019. ISSN 22128689. Disponível em: <<https://linkinghub.elsevier.com/retrieve/pii/S2212868917300995>>.

SAPOUNIDIS, T.; DEMETRIADIS, S.; STAMELOS, I. Evaluating children performance with graphical and tangible robot programming tools. **Personal and Ubiquitous Computing**, v. 19, n. 1, p. 225–237, 2015. ISSN 1617-4909, 1617-4917. Disponível em: <<http://link.springer.com/10.1007/s00779-014-0774-3>>.

SCUPIN, R. The KJ Method: A Technique for Analyzing Data Derived from Japanese Ethnology. **Human Organization**, v. 56, n. 2, p. 233–237, jun. 1997. ISSN 0018-7259. Publisher: Society for Applied Anthropology. Disponível em: <<https://sfaajournals.net/doi/10.17730/humo.56.2.x335923511444655>>.

SHEEHAN, K. J.; PILA, S.; LAURICELLA, A. R.; WARTELLA, E. A. Parent-child interaction and children's learning from a coding application. **Computers & Education**, v. 140, p. 103601, out. 2019. ISSN 03601315. Disponível em: <<https://linkinghub.elsevier.com/retrieve/pii/S036013151930154X>>.

SILVA, M.; ROBERTO, R.; TEICHRIEB, V. Evaluating an educational system based on projective augmented reality. In: . [s.n.], 2013. Disponível em: <<http://www.br-ie.org/pub/index.php/sbie/article/view/2499>>.

SIPITAKIAT, A.; NUSEN, N. Robo-Blocks: designing debugging abilities in a tangible programming system for early primary school children. In: **Proceedings of the 11th International Conference on Interaction Design and Children - IDC '12**. Bremen, Germany: ACM Press, 2012. p. 98. ISBN 978-1-4503-1007-9. Disponível em: <<http://dl.acm.org/citation.cfm?doid=2307096.2307108>>.

- SNOW, C. E.; BURNS, M. S.; GRIFFIN, P. **Preventing reading difficulties in young children**. Washington, DC: National Academy Press, 1998. ISBN 978-0-309-06418-7.
- SOLOMON, C.; HARVEY, B.; KAHN, K.; LIEBERMAN, H.; MILLER, M. L.; MINSKY, M.; PAPERT, A.; SILVERMAN, B. History of Logo. **Proceedings of the ACM on Programming Languages**, v. 4, n. HOPL, p. 1–66, jun. 2020. ISSN 2475-1421, 2475-1421. Disponível em: <<https://dl.acm.org/doi/10.1145/3386329>>.
- SULLIVAN, A.; ELKIN, M.; BERS, M. U. KIBO robot demo: engaging young children in programming and engineering. In: **Proceedings of the 14th International Conference on Interaction Design and Children - IDC '15**. Boston, Massachusetts: ACM Press, 2015. p. 418–421. ISBN 978-1-4503-3590-4. Disponível em: <<http://dl.acm.org/citation.cfm?doid=2771839.2771868>>.
- SUTHERLAND, I. E. A head-mounted three dimensional display. p. 8, 1968.
- VALDIVIA-GARCIA, H. **Understanding the Impact of Diversity in Software Bugs on Bug Prediction Models**. Thesis — Rochester Institute of Technology, 2016. Disponível em: <<https://scholarworks.rit.edu/theses/9305>>.
- VALENTE, J. A. Aspectos críticos das tecnologias nos ambientes educacionais e nas escolas. v. 2, n. 3, p. 11–28–28, 2018. ISSN 2238-1279. Number: 3. Disponível em: <<http://periodicos.estacio.br/index.php/reeduc/article/view/4891>>.
- VALKENBURG, J. JOINING THE CONVERSATION: Scaffolding and Tutoring Mathematics. v. 15, n. 2, p. 9, 2010.
- VIANA, C.; RAABE, A. Interface de programação tangível para produção de algoritmos sonoros. In: . Fortaleza, Ceará, Brasil: [s.n.], 2018. p. 125. Disponível em: <<http://br-ie.org/pub/index.php/wcbie/article/view/8219>>.
- VIANA, C. P. **Interface híbrida para o brinquedo de programar RoPE**. Tese (Trabalho Técnico-científico de Conclusão de Curso) — Universidade do Vale do Itajaí, Itajaí, 2018.
- WAZLAWICK, R. S. **Metodologia da Pesquisa para Ciência da Computação**. 6. ed. Rio de Janeiro: Campus, 2009. ISBN 978-85-352-6643-6.
- WEINTROP, D. Block-based programming in computer science education. **Communications of the ACM**, v. 62, n. 8, p. 22–25, jul. 2019. ISSN 0001-0782, 1557-7317. Disponível em: <<https://dl.acm.org/doi/10.1145/3341221>>.
- WING, J. Blog, **COMPUTATIONAL THINKING BENEFITS SOCIETY**. 2014. Disponível em: <<http://socialissues.cs.toronto.edu/index.html%3Fp=279.html>>.
- WING, J. M. Computational Thinking. **COMMUNICATIONS OF THE ACM**, v. 49, n. March, p. 3, 2006. Disponível em: <<https://www.cs.cmu.edu/~.15110-s13/Wing06-ct.pdf>>.
- _____. Computational thinking and thinking about computing. **Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences**, v. 366, n. 1881, p. 3717–3725, out. 2008. Publisher: Royal Society. Disponível em: <<https://royalsocietypublishing.org/doi/10.1098/rsta.2008.0118>>.
- WONG, G. K.; JIANG, S. Computational thinking education for children: Algorithmic thinking and debugging. In: **2018 IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE)**. IEEE, 2018. p. 328–334. ISBN 978-1-5386-6522-0. Disponível em: <<https://ieeexplore.ieee.org/document/8615232/>>.
- WRZESIEN, M.; BOTELLA, C.; BRETÓN-LÓPEZ, J.; GONZÁLEZ, E. del R.; BURKHARDT, J.-M.; ALCANIZ, M.; PÉREZ-ARA, M. Treating small animal phobias using a projective-augmented reality system: A single-case study. v. 49, p. 343–353, 2015. ISSN 0747-5632. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S074756321500093X>>.

YU, J.; ROQUE, R. A review of computational toys and kits for young children. **International Journal of Child-Computer Interaction**, v. 21, p. 17–36, set. 2019. ISSN 22128689. Disponível em: <<https://linkinghub.elsevier.com/retrieve/pii/S2212868918300825>>.

ZUCKERMAN, O.; GAL-OZ, A. To TUI or not to TUI: Evaluating performance and preference in tangible vs. graphical user interfaces. **International Journal of Human-Computer Studies**, v. 71, n. 7-8, p. 803–820, jul. 2013. ISSN 10715819. Disponível em: <<https://linkinghub.elsevier.com/retrieve/pii/S1071581913000438>>.

ÇİFTÇİ, S.; BILDİREN, A. The effect of coding courses on the cognitive abilities and problem-solving skills of preschool children. **Computer Science Education**, v. 30, n. 1, p. 3–21, jan. 2020. ISSN 0899-3408, 1744-5175. Disponível em: <<https://www.tandfonline.com/doi/full/10.1080/08993408.2019.1696169>>.

APÊNDICE A – STRINGS DE BUSCA PARA REVISÃO SISTEMÁTICA DA LITERATURA

Base	String
ACM Digital Library	<i>"query": AllField:((children OR kindergarten OR "childhood education") AND (activities OR "pedagogic practices" OR "curriculum") AND (robotics OR "programmable toy" OR "computational toy") AND ("computational thinking") AND (algorithm* OR program*)) AND Title:(NOT (survey OR secondary OR middle OR "high school")) "filter": Article Type: Research Article, Publication Date: (01/01/2015 TO 06/30/2020), ACM Content: DL</i>
ERIC	<i>((children OR kindergarten OR "childhood education") AND (robotics OR "programmable toy" OR "computational toy") AND ("computational thinking")) Tags: "Since 2016", "Early Childhood Education", "Journal Articles"</i>
IEEE Xplore	<i>("Full Text .AND. Metadata":children OR "Full Text .AND. Metadata":kindergarten OR "Full Text .AND. Metadata":"childhood education") AND ("Full Text .AND. Metadata":activities OR "Full Text .AND. Metadata":"pedagogic practices" OR "Full Text .AND. Metadata":"curriculum") AND ("Full Text .AND. Metadata":robotics OR "Full Text .AND. Metadata":"programmable toy" OR "Full Text .AND. Metadata":"computational toy") AND ("Full Text .AND. Metadata":"computational thinking") AND NOT ("Document Title":"high school" OR "Document Title":"survey" OR "Document Title":"secondary education" OR "Document Title":"undergraduate" OR "Document Title":"middle" OR "Document Title":"teacher")</i>
ScienceDirect	<i>Terms:((children OR kindergarten OR "childhood education") AND (robotics OR "programmable toy" OR "computational toy") AND ("computational thinking")) Years: 2015-2020</i>

APÊNDICE B – REQUISITOS DO APLICATIVO

Os requisitos são histórias de usuário conforme o modelo: *Eu, como [papel], (preciso|gostaria) [funcionalidade] para [benefício]*. A vantagem deste modelo é que ele mostra claramente quem precisa da funcionalidade e qual a justificativa.

- RF01 – Eu, como professor, gostaria que o sistema não listasse dispositivos além dos que são brinquedos RoPE, para facilitar a seleção.
- RF02 – Eu, como professor, gostaria que o celular retornasse para a tela inicial caso a conexão seja perdida, para que seja possível reconectar.
- RF03 – Eu, como criança, preciso que a sequência de blocos de papelão seja transferida ao RoPE como um algoritmo, para que possa programá-lo sem tocar no celular.
- RF04 – Eu, como criança, preciso ver projetado no chão uma marcação indicando quais blocos fazem parte do algoritmo que estou construindo, para saber quais blocos serão executados e quais não serão executados.
- RF05 – Eu, como criança, preciso que os comandos executados pelo brinquedo sejam destacados, para que eu saiba qual bloco corresponde a qual movimento do brinquedo.
- RF06 – Eu, como pesquisador, preciso que os programas criados pelas crianças estejam disponíveis para consulta posterior, para que sirva como insumo de análise.
- RF07 - Eu, como professor/pesquisador, gostaria de ativar a execução passo a passo (depuração), para a criança perceber o efeito de cada comando.

Já os requisitos não funcionais não estão relacionados a serviços, mas sim a propriedades de segurança, confiabilidade e tempo de resposta. Além disso, os requisitos não funcionais podem afetar a arquitetura geral de um sistema.

- RNF1 – Eu, como pesquisador, preciso que o sistema funcione em dispositivos com Android 6.0, para suportar ao menos 70% dos dispositivos Android.
- RNF2 – Eu, como criança, preciso que a luminosidade do projetor seja suficiente para eu visualizar a projeção em ambientes claros a uma distância de 1 metro entre a lente e a superfície de projeção, para possibilitar o uso em salas de aula.

- RNF3 - Eu, como criança, preciso que o tempo de detecção e destaque dos blocos seja menor que 500ms, para não prejudicar a usabilidade.



TERMO DE ASSENTIMENTO LIVRE E ESCLARECIDO

Você está sendo convidado(a) para participar, como voluntário(a), em uma pesquisa sobre uma forma de programar um brinquedo programável com blocos de papelão e iluminação com projetor. Caso quiser participar, continue lendo os procedimentos da pesquisa.

Depois que eu passar a você as informações da pesquisa, **será necessário que você responda algumas perguntas sobre os blocos de papelão, programe o brinquedo utilizando esses blocos para andar sobre uma sequência de mapas, e encontre erros em programas existentes.** Se não quiser participar não há problema algum, você não precisa me explicar o motivo. É seu direito não participar do estudo.

Título do estudo: **Interface de Realidade Aumentada para o Brinquedo Programável RoPE.**

Os brinquedos programáveis são pequenos brinquedos com rodas, que guardam os comandos que crianças informam, e depois executam cada comando em sequência, se movendo e produzindo sons e luzes. Esses brinquedos buscam estimular o raciocínio de crianças entre 3 e 8 anos.

Existem muitas formas de programar esses brinquedos, como botões e aplicativos de celular. Essa pesquisa testa a programação por meio de blocos de papelão com auxílio de um projetor, com o **objetivo de verificar como a visão da estrutura e da execução do algoritmo por meio de projetor pode influenciar o aprendizado.** As atividades realizadas serão:

1. Responder o que entendeu ao ver os desenhos de cada um dos blocos
2. Programar o brinquedo usando os blocos
3. Encontrar erros em programas existentes

Caso participe do estudo, estará fornecendo dados para contribuir com o avanço das técnicas de ensino de programação. **Um aplicativo de celular irá guardar os programas criados e irá gravar um vídeo da atividade.** Com isso poderemos analisar as falas e movimentos corporais feitos durante o experimento e verificar se essa técnica facilita que crianças compreendam o que é um programa e saibam encontrar erros nesses programas.

Os resultados dessa pesquisa serão divulgados através de produção de dissertação para conclusão do curso de Mestrado. Porém, a **sua identidade será preservada e não divulgada junto aos resultados.** Esses resultados vão ajudar pesquisadores e professoras a criar atividades para o desenvolvimento do raciocínio lógico de crianças, melhorando as formas de programar brinquedos programáveis. **Possíveis riscos são querer continuar brincando, ficar com vergonha de concluir as tarefas e ter medo de errar.** Você não será avaliado, e os erros não são ruins. Não precisa ter medo de errar e você não será



repreendido por isso. **Danos acidentais aos equipamentos usados no experimento também não são sua responsabilidade.**

A participação neste experimento é voluntária, portanto **não há remuneração**. Em caso de qualquer prejuízo causado pela participação no estudo, haverá ressarcimento conforme indicado por lei.

Os resultados coletados serão armazenados em meio digital por até 5 anos. A qualquer momento você pode retirar a autorização de participação bem como exigir a exclusão das informações coletadas.

Deixo meu contato de celular (47) 99629-1949 e e-mail (cesarviana@edu.univali.br) para quaisquer esclarecimentos. Dúvidas também podem ser retiradas diretamente com o Comitê de Ética da Univali, pela central de atendimento (0800 723 1300), telefone ((47) 3341-7738) ou e-mail (etica@univali.br).

Caso persistam dúvidas, sugestões e/ou denúncias após os esclarecimentos do pesquisador o Comitê de Ética (CEP) da Universidade do Vale do Itajaí – UNIVALI está disponível para atender.

CEP/UNIVALI - Rua Uruguai, n. 458 Centro Itajaí. Bloco F6, andar térreo.

Horário de atendimento: Das 8:00 às 12:00 e das 13:30 às 17:30

Telefone: (47) 3341-7738. E-mail: etica@univali.br

ASSENTIMENTO DE PARTICIPAÇÃO

Eu, _____, abaixo assinado, concordo em participar do presente estudo como participante. O pesquisador me falou sobre tudo o que vai acontecer na pesquisa, o que terei que fazer, inclusive sobre os possíveis riscos e benefícios envolvidos na minha participação. O pesquisador me garantiu que eu poderei sair da pesquisa a qualquer momento, sem dar nenhuma explicação.

Local e data: _____

Nome: _____

Assinatura do Participante ou Responsável: _____

Telefone para contato: _____

Nome do Pesquisador Responsável: _____

Telefone para contato: (47) 99629-1949 | Email: cesarviana@edu.univali.br

APÊNDICE C – MÓDULOS DO APLICATIVO

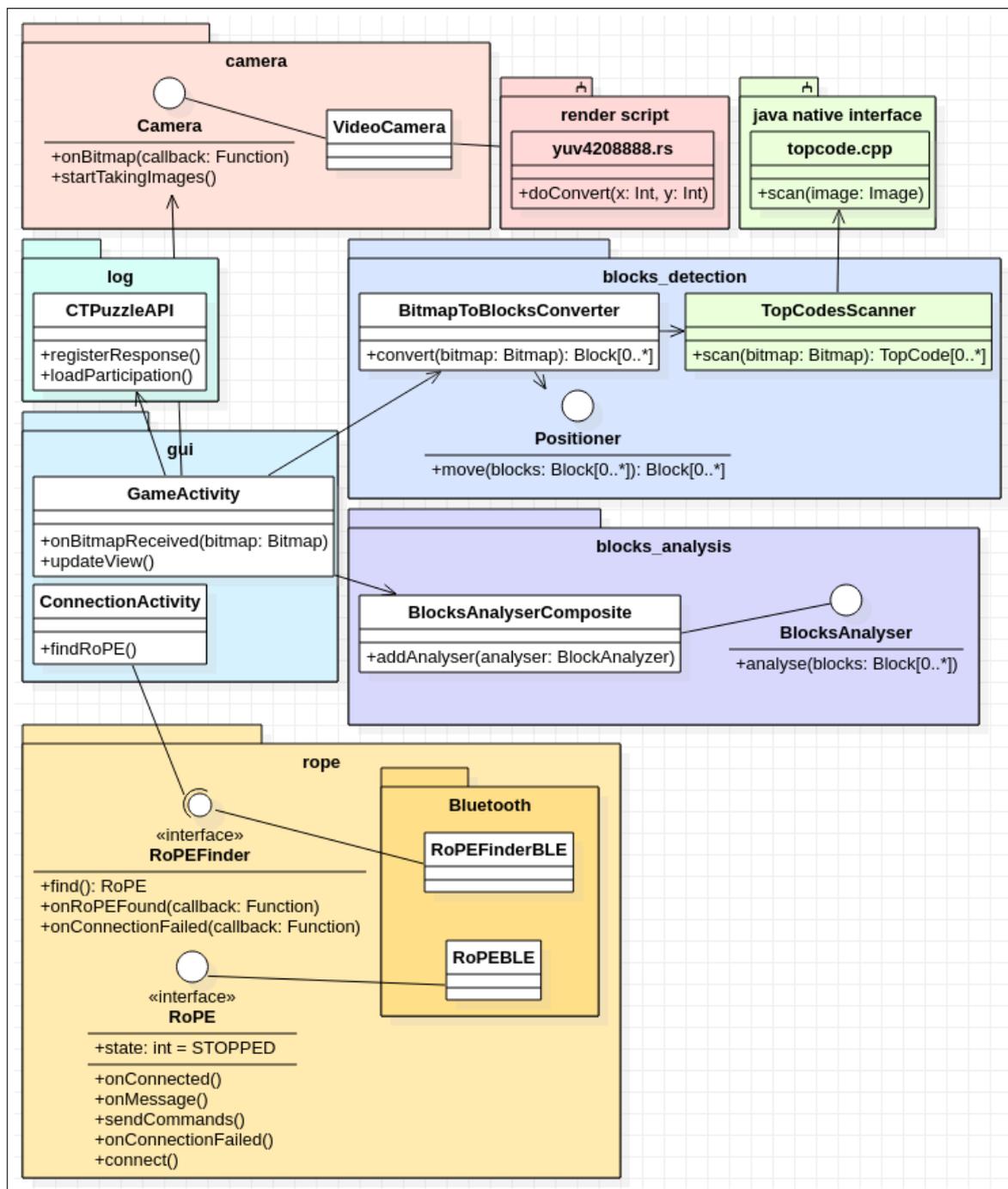


Figura 1. Módulos do aplicativo de realidade aumentada.

A organização em módulos facilita o reuso do código por trabalhos futuros. Um módulo possivelmente útil é o de comunicação com o RoPE, que abstrai a complexidade de utilização do Bluetooth. Duas tecnologias essenciais para a execução do projeto foram o Render Script e a *Java Native Interface* (JNI). O Render Script foi utilizado para converter imagem produzida pela câmera

de formato YUV para RGB, exigido pela biblioteca Topcodes. O JNI permitiu invocar código nativo (escrito em C++), o que diminuiu o tempo de identificação dos topcodes.

Os módulos de detecção de blocos e análise de blocos se diferenciam pelo que o primeiro recebe uma imagem e transforma em objetos tratados pelo aplicativo, como blocos, marcas de calibragem e objeto RoPE. O módulo de análise possui um conjunto de analisadores, que observam o posicionamento desses objetos e emitem eventos. Um exemplo de analisador é o `ProgramDetector`, que observa se há uma sequência de blocos encaixados formando um algoritmo e então notifica o controle principal, que envia o algoritmo ao RoPE.

APÊNDICE D – PROTOTIPAÇÃO

A prototipação validou o uso do mini-projetor UC46. Este possui uma potência de 1200 lumens, e havia preocupação quanto à qualidade da imagem gerada. O teste ocorreu em uma sala de 25 metros quadrados, com uma lâmpada de LED, de 7W e luminosidade de 7 lumens, e uma janela parcialmente aberta (Figura 2a). A incidência de luz externa (porta aberta) sobre a projeção prejudicou a qualidade da imagem, porém com a porta fechada o contraste obtido viabilizou a continuidade do projeto. A



(a) Projeção em ambiente iluminado por lâmpada de 700 lumens (b) Projeção com a incidência de luz externa (c) Projeção sem incidência de luz externa

Figura 2. Qualidade da projeção em diferentes condições luminosas.

Fonte: O autor.

estrutura para posicionar o projetor usa PVC e não utiliza cola, para permitir reprodução. O projeto da estrutura foi feito em 3D e busco eliminar ao máximo o uso de materiais. A rigidez da estrutura também foi testada para verificar se não cairia para os lados.

A prototipação também permitiu corrigir o posicionamento da imagem captada pela câmera. Devido à inclinação entre a câmera e a superfície de projeção, a visão da câmera fica em perspectiva. Porém essa perspectiva precisa ser eliminada para encontrar a posição correta dos blocos no mapa (Figura 3). Isso é feito criando uma matriz homográfica (DUBROFSKY, 2009).

Protótipos dos blocos de papelão (Figura 4) produzidos tem cores correspondentes às cores dos botões do brinquedo RoPE. O tamanho é 7,5x10cm e o diâmetro da marca fiducial é 2,5cm, de modo que a câmera reconhece os blocos a 100cm de distância. Posteriormente o diâmetro foi aumentado para 4cm.

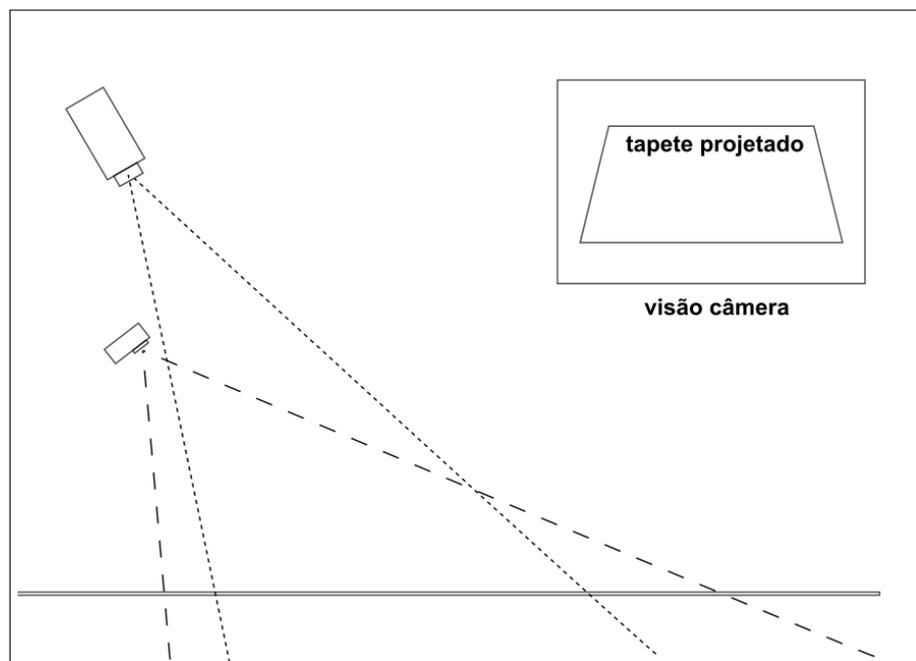


Figura 3. Inclinação entre câmera, projetor e superfície.



Figura 4. Blocos de papel e projeção de um indicador para o local do próximo bloco.

APÊNDICE E – MINI-PROJETOR



Figura 5. Mini-projetor UC46.

Fonte: Amazon.com

Especificação	Valor
Lumens	1200
Dimensões	20.83 x 16 x 7.87cm
Peso	952.54 g
Conexões	HDMI, USB, Cartão SD, VGA
Tamanho projeção	Até 130 polegadas

APÊNDICE F – ENTREVISTAS COM AS PROFESSORAS

Dia 1

[Pesq] Vocês poderiam falar um pouco sobre o trabalho de vocês, quantos anos são professoras, como é a atividade aqui, pra que eu entenda um pouco e possa conhecer.

[Prof_1] Uhum. Ah...

[Pesq] É informal.

[Prof_1] Tá. Eu vou falando e você vai me perguntando.

[Pesq] Uhum, sim.

[Prof_1] Eu trabalho na rede a 8 anos, estou no 8º ano, sou formada em pedagogia, tenho pós graduação em Educação Anos Iniciais e Educação Infantil, e.. essa turma aqui é dos 5 a 6 anos, é a primeira vez que pego essa turma aqui mas eu já trabalhei com todas as turmas, todas as idades. Eu gosto muito, eu me identifico muito, eu me identifico com a turma do pré. Porque daí é o letramento...

[Pesq] Então já começa trabalhar com o alfabeto, isso?

[Prof_1] É.. Não é o alfabeto, é uma introdução né, a alfabetização que a gente chama é só lá na escola

[Pesq] Só pode conhecer as letras assim...

[Prof_1] Pode, reconhecer, escrever o nome... cores.

[Pesq] Você já tem experiência a bastante tempo né, o que você pensa sobre as crianças usando tecnologia?

[Prof_1] Eu acho bom, muito interessante. No meu TCC eu fiz ludicidade na Educação Infantil, então a tecnologia é bem importante. É interessante introduzir isso.

[Pesq] E como você acha que ela pode contribuir ou não contribuir pro desenvolvimento da criança?

[Prof_1] Eu penso que sim, porque é uma forma educativa de inserir eles na informatização, porque em casa o celular e a tecnologia é só voltada pra YouTube, coisas que não são tão úteis. E se a

gente puder introduzir isso de uma forma que eles procurem no Google o conhecimento, informática.. isso é bem legal. Interessante. Mudar o foco do YouTube...

[Pesq] E qual a importância que você daria para as crianças estarem vendo as coisas. Por exemplo, se eu vou tentar explicar alguma coisa só falando será que isso é suficiente ou será que eu mostrar pra criança uma imagem.. de que modo isso influencia no aprendizado dela.

[Prof_1] Ah, eu acho importantíssimo, porque eu sou visual, eu aprendo muito mais olhando e ouvindo, presencial. Tirar dúvidas né, na hora.

[Pesq] E na questão dos materiais, se vou tentar explicar alguma coisa só falando ou tendo algum objeto que eu possa apontar e querer explicar algo mostrando aquilo pra criança.

[Prof_1] Ah sim, é a ludicidade né, tem que mostrar, aprende muito mais.

[Pesq] É isso que estou tentando né. O robô pode ser programado usando os botões, mas a criança não vê o que está sendo programado. Ela pode apertar várias vezes os botões e esquecer. E aí, usando os blocos, o que estou tentando trazer..

[Prof_1] É na prática né..

[Pesq] .. é que ela tá vendo..

[Prof_1] Não, isso é fundamental, eu acho bem interessante.

[Pesq] Pensando neste projeto específico, o que você achou ele? A respeito da aplicabilidade, se fosse pra ser usado aqui.. ou em alguma outra sala de aula, será que seria usável, pra alguma atividade semelhante, usando essa ideia de projetar a imagem e ter um robzinho se mexendo...

[Prof_1] Eu acho bem legal, porque é convidativo para eles assim. Isso poderia ser usado como se a gente colocasse um reto-projetor pra eles?

[Pesq] Sim, a diferença é...

[Prof_1] Tu tá encontrando ali pelo celular.

[Pesq] Tá no celular, só que a criança não interage com o celular, ela interage com o celular.

[Prof_1] E com a imagem..

[Pesq] E com a imagem. Quando o robô chega naquela posição o celular entende e daí ele dá um feedback, faz um som do robô comendo a maçã. Mas isso poderia ser evoluído, por exemplo, passou em cima de uma ponte, faz o barulho da ponte, ou chegou no número 3, fala "chegou no numero 3", ou se tem um obstáculo faz o robô parar, porque não pode passar ali.

[Prof_1] É como um jogo né.

[Pesq] Uhum.

[Prof_1] É um jogo... bem legal, de comandos né.

[Pesq] E alguma sugestão que você daria que pudesse melhorar o projeto, ou a ideia. ou alguma coisa que você já viu, que pudesse ser mais útil pra vocês e pro desenvolvimento da criança. Teria alguma sugestão, "ah muda isso".

[Prof_1] Ah, não sei dizer, isso é novo pra mim... eu nunca tinha visto isso...

[Pesq] Estou tentando trazer uma proposta nova...

[Prof_1] Já tinha visto Prof_2?

[Prof_2] Não, isso é totalmente novo pra gente. Isso é totalmente novo, acredito que daqui mais um tempo vai ser super necessário e a gente torce pra que isso realmente acontece porque hoje a tecnologia está influenciando todas as áreas e acho que a educação tanto infantil como anos iniciais.. tem que ser tudo tecnológico hoje.

[Pesq] Eu imagino pro desenvolvimento depois de adulto..

[Prof_2] É, deixa te contar uma história. Estou na faculdade também, estou no último período de pedagogia. E quando eu comecei fazer os estágios eu penei muito, até hoje. Eu vou fazer um curso, nem que seja básico de informática, pra estar, ali pelo menos pra conhecer algumas coisas.

[Pesq] Edição de...

[Prof_2] Exatamente... Então eu acho isso muito importante, tem que ter inserido já bem antes

[Pesq] E o desenvolvimento do raciocínio lógico também.. das direções... A parte que eu desenvolvi foi essa ideia da projeção e da comunicação com o celular. Mas o robô já existem a bastante tempo, e é usado em outras creches em Balneário e Itajaí, só o robô. As crianças adoram.

[Prof_2] Imagina.

[Pesq] A minha parte foi uma parte pequena no projeto, e estou testando essa nova abordagem, isso que estou tentando avaliar. Minha parte foi só um pedacinho.

[Prof_2] Mas que não deixa de ser importante né.

Dia 2

[Pesq] São umas perguntas bem gerais, só para eu conhecer do trabalho, porque a pesquisa é qualitativa, então eu tenho que entender não só sobre o projeto, mas também sobre o contexto geral. Primeiro eu gostaria de perguntar se posso gravar isso porque depois é melhor..

[Prof_3] É melhor pra transcrever né, pq se a gente for anotando perde muita coisa né.

[Pesq] Eu me acostumei muito a digitar, então só pra escrever (no papel)... Então. Eu gostaria que você falasse sobre o teu trabalho.. Quanto tempo tem contato com as crianças, eu percebi que é bastante tempo né, que tu trabalha com crianças.

[Prof_3] Sim, aham. Bom eu já trabalho com crianças desde 1992. Comecei só na Educação Infantil, e aí depois na Secretaria da Educação, com a Educação Infantil, coordenando todas as... e aí a gente começou o processo de fazer uma proposta pedagógica, que até então não tinha, então a gente conduziu, fez formação, tudo pra montar a proposta né. E depois eu saí da Secretaria, voltei pra sala de aula. Desde 2014 então eu estou em sala de aula, com as várias turmas. Então a gente planeja,.. a gente usava a proposta pedagógica, que é as Linguagens, Brincadeiras e Interações, e depois quando teve agora a BNCC a gente tá seguindo os campos de experiências, os 5 campos de experiências.

[Pesq] Quando que entrou em vigor essa parte da BNCC.

[Prof_3] Oficialmente foi ano passado, então a gente teve que seguir a proposta, porque a gente usou bastante documentos do MEC também pra elaborar né. Então ela é próxima à BNCC, só que ela trabalhava mais com os conceitos né, e a BNCC trabalha mais com os campos de experiência amplos e com os objetivos que ali tem né... por faixa etária né...

[Pesq] A BNCC não fala como tem que ser feita a atividade...

[Prof_3] Não, ela tem objetivos e a gente, como professora, tem a liberdade de escolher qual a atividade que mais próxima ou mais no contexto né. Então, como a gente aqui construiu os cantos,

todos esses cantos que tem aqui foram ideias das crianças né. A BNCC fala dos espaços como forma educativa também né. Então a gente pega das crianças pra ampliar, então todos esses espaços são sugestões que eles deram. Então aqui no robô, que eles queriam um robô. E aí eu disse "mas como a gente vai fazer um robô" e eles "ah, a gente vai fazer com um controle remoto", e disse "mas a gente não tem essa tecnologia, como a gente podia fazer?". Aí um disse "ah, coloca fios, tem que colocar fios" e eu disse "mas a prof não entende de elétrica".

[Pesq] Eu lembro que ontem eu tirei a extensão da mochila e o menino falou "olha quanto fio!".

[Prof_3] Aham. Então eles entendem que há uma coisa de funcionamento né, que tem que ter fios, que tem que ter... mas aí a gente construiu então de sucatas, pra guardar os carrinhos, porque esse canto aqui eles gostam muito que tem os carrinhos né, tem as pistas que a gente ganhou de doação, então eles amam isso. A gente fez pra organizar mais né. Então ele não tem uma funcionalidade assim, remota né.

[Pesq] Mas ele já entendem que...

[Prof_3] Sim, eles já entendem que existe uma tecnologia. Então eu trouxe pra eles também um vídeo sobre os robôs né.. eles assistiram, então a gente vai ampliando, traz do mundo deles pra ampliar um pouco mais.

[Pesq] Isso era um filme, eu sei que tem aquele filme dos robôs.

[Prof_3] Não, é mais tecnológico assim, a gente trouxe robôs de verdade.

[Pesq] Tu já falou um pouco sobre isso né, mas o que tu pensa sobre as crianças tendo contato com tecnologia em geral?

[Prof_3] Eu acho fantástico, é uma pena que nós não temos né. Na rede pública é mais difícil porque ter mais equipamentos né, de pesquisa com as crianças, como uma mesa digital, isso é muito custoso né, pro município ter. A gente sabe que nem temos espaço pra isso, tem que ter uma sala especial porque os materiais são caros e não dá pra ficar aqui e ali, se perdem e não tem contato. Mas isso é interessante trazer né, porque outra pessoa que tenha o entendimento isso eu achei muito legal assim, porque eles gostaram assim, eles ficaram vidrados naquele robozinho. "E como é que comeu a maçã" eu achei engraçado né ele olhou por baixo onde é que vai a maçã né. Mas eu acho que ainda é um pouco distante. Talvez né daqui mais algum tempo, quando isso ficar mais acessível. Porque eu sei que um laboratório, informática e tudo, digital, isso é um investimento alto né, então fica mais

nas universidades. Eu sei que na Furb, a gente foi lá ver essa mesa digital. É muito legal. Porque nós professores, nós não temos a formação tecnológica né. Então pra gente lidar com o computador... Agora o município nos disponibilizou um computador. Ele é um chrome book. Só que nós não sabemos lidar. Eu fiquei no meu computador mesmo, porque ele não tem o pacote windows. Então, como é que a gente, que mal sabe mexer naquele, vai aprender a mexer com esse tipo.. Que vai ter que armazenar em outro tipo de formato... aí como vou fazer os vídeos, porque a gente faz vídeo, a gente faz vídeo, a gente elabora várias coisas né. Eu já fazia antes, sempre usei um pouco de vídeo com as crianças né, filmava e tudo mais. Mas agora com a pandemia, foi muito mais necessário tá disponibilizando isso né. Porque quando eles estavam em "home-office", a gente fazia vídeos pra explicar o que era a brincadeira. Pra eles nos verem, pra lembrar de nós. Então, tudo isso pra, isso tudo é difícil pra gente pegar esse fio da meada, de montar né...

[Pesq] É complexo né...

[Prof_3] É difícil, e demora porque quando você vai editar né. Porque quando você vai editar... A edição leva muito tempo.. então cortar.

[Pesq] Então eles deram um computador pra cada professora?

[Prof_3] Pra cada professora.

[Pesq] Mas não deram formação pra criança de...

[Prof_3] Sim, prometeram que vão dar uma formação. Então eu só não devolvi o notebook pra eles, porque a gente assina um termo que se acontecer alguma coisa nós temos que pagar. Então, eu vou esperar a formação, pra ver se eu consigo... se não for vantajoso eu vou devolver. Então esse tipo de coisa é bastante complicado né. Então eu que mais ou menos né.. sou meio burrinha na questão de... gosto muito de tecnologia, uso bastante, mas ainda não tenho toda... imagina quem tem mais dificuldade. Porque teve professoras que tinham muita dificuldade, então isso é complicado né. Então eu acho que na formação do professor hoje em dia, na formação universitária, tem que ter a formação de tecnologia. Tem que ter na grade, porque isso é imprescindível. A gente viu agora né, nesses tempos, que é importante a gente ter esse tipo de formação na própria formação do professor inicial.

[Pesq] Uma coisa que você falou né, de ter um laboratório de informática e tal né.. eu li um artigo, sobre o RoPE, em que ai idéia é inverter essa lógica do laboratório de informática. Vamos dizer, uma escola, tem 45, 50 minutos de aula. O professor ele quer fazer uma atividade no laboratório de informática, até todo mundo sentar, ir e voltar meia hora de aula perdida. E aí ela fala que a ideia do

RoPE é inverter, trazer a informática pra dentro da sala. Claro que esse negócio que eu fiz não é tão portátil assim, mas o robô em si ele é pequeno. Podia ter três, quatro aqui e as crianças brincando. Claro que não é "oh", um computador né que a criança faz tudo.

[Prof_3] Mas é legal isso mesmo, tirar um pouco essa questão de ficar na máquina sentado e olhando pra algo. Ter um concreto no caso, ele é tipo, ele é 4D, ele tá ali junto com as crianças. (Diferente de) que ficar clicando ali na setinha né. Isso é interessante, pros menores ia ser interessante também. E pros maiores seria como o mecanismo que isso acontece né.

[Pesq] Por dentro ali né?

[Prof_3] Ahan.

[Pesq] Na verdade até a gente aprende com o robô lá na universidade. Eu tô aprendendo, to mexendo dentro dele ali, essa semana deu um erro que não conseguia gravar mais nada nele e tive que ir lá em balneário, pedir ajuda pro Eng_0, que foi o cara que projetou, e ele me mostrou um monte de coisa, como fazia né, porque já tinham tido esse mesmo erro. Uma vez ficaram um mês, vi eles batendo cabeça e daí ele resolveu. Eu não tenho um mês. Eu também to aprendendo, só que um nível mais pra dentro do robô.

[Prof_3] O legal né, como eu tava falando pra você né. Pros menores é interessante eles seguirem um circuito completo, como eu falei pra você né. Eles vão seguir aquele roteiro pra poder encontrar, pra poder fazer todo o trajeto. É melhor pro entendimento deles. E aí então, pros pequenos... Porque no caso você tá fazendo essa pesquisa com todos... só a partir de quatro anos.

[Pesq] Eu to testando, na verdade, a ideia de quatro a seis. Mas é testar essa abordagem né. Talvez pra quatro não seja interessante, ou será que ajuda a criança mais velha entender, ou mais nova...

[Prof_3] Ou trazer várias facetas, na mesma atividade vários níveis.

[Pesq] No caso o robô em si ele é programado pelos botões originalmente, e tem a teoria de design que fala que as coisas tem que estar visíveis. E o que eu vejo é que nem sempre o programa está visível por meio dos botões, principalmente pra programas mais longos. Então a criança vai lá, apertou vários botõezinhos, então ela esquece aquilo, ou ela não consegue alterar. Então a ideia dos blocos é permitir que isso se torne mais visível. O programa, é um algoritmo que a gente fala, é uma sequência de passos. A única diferença é que ela tá visível por meio dos blocos, tanto a estrutura, que é a sequência de um bloquinho depois do outro, quanto a execução, que seria "esse bloco que está

executado agora".

[Prof_3] Ah tá, porque mais fácil de fazer a leitura. Aquela leitura que coloca quando você junta a programaçãozinha aqui embaixo ele já lê e já executa né.

[Pesq] Isso, pra criança conseguir perceber né. Aquelas marquinhas redondinhas, o ideal é que nem tivesse aquilo. Eu tive que usar por questão de tecnologia, pra conseguir implementar né. Então a ideia é a criança conseguir entender que o algoritmo é aquela sequência de blocos, e qual que está sendo executado.

[Prof_3] Mas eu acho que foi bem legal assim, mesmo tendo assim alguns que são menores, mas eu acho que deu pra eles estarem fazendo né, eu acho que foram bem assim (risos).

[Pesq] Eu também achei. O Menino_M achei que não ia conseguir e conseguiu.

[Prof_3] Conseguiu? Ah que bom, que ele tem bastante dificuldade de entendimento assim. Mas que bom.

[Pesq] Tem mais duas perguntas aqui. A importância que tu dá pra questão da visibilidade, das coisas estarem visíveis pra criança conseguir pegar na mão, o quão importante isso é pra criança entender...

[Prof_3] Ah, sim. Pros menores, até a criança de sete, oito anos, é sempre concreto né, é sempre a partir daquele material, que ele possa pegar, visualizar, pra ele internalizar. Porque o abstrato assim, ficar explicando muito, eles não vão conseguir fazer a associação. Eles precisam associar que uma coisa é ligada a outra. Então eles tem que sentir, eles tem que ver. Então essa forma ali eles conseguiram ter os comandos na hora. E no computador ele vai ter só uma setinha, que ele vai né, e vai colocar né, tipo clicar. E aqui não, ele vai ter que fazer algo, teve que construir algo e entender aquilo pra depois programar né.

[Pesq] Eu vi que tem uns pedacinhos de madeira ali né, perguntei pra Prof_6, sobre o quela aquilo né, ela falou que é sobre o contato com a natureza.

[Prof_3] Isso, é um outro elemento. A gente está acostumado com plástico. Então plástico, madeira, são outras possibilidades de construção. Esses dias ele tava fazendo um prédio, disse que tava fazendo um prédio. Ele empilhou todos os toquinhos assim. Então é isso mesmo, é construir ou construir uma cerca. O bloco é essa tentativa. Ele não é um objeto muito estruturado. A criança tem que ter contato com objetos não estruturados. Então ele não tem forma, ele não deve pra um devido

fim. Ele vai montar uma estrutura a partir daquele objeto que não tem formato. Ele que vai ter que imaginar e dar forma aquele objeto. Então o objeto não estruturado é imprescindível que tenha no espaço de educação.

[Pesq] Um objeto que não foi decidido que...

[Prof_3] É, ele não tem uma função. Então, "ah, isso aqui é um jogo" então tu vai completar assim. Ele é um material que dá pra fazer mil coisas.

[Pesq] Algo aberto...

[Prof_3] Isso, aberto.

[Pesq] Então no caso essa atividade que trouxe tá um pouco estruturada né.

[Prof_3] É estruturada, tem objetivos e regras. Tens que montar primeiro, ver a sequência ali, então pra ele estrutura isso... por isso que os menores não tem essa... eles vão lá, vão brincar né, os menores. Os maiores vão entender que tem uma regra, se só pode apertar "esse botão", se "só pode seguir esse trajeto", "não pode fazer diferente". Então ele tem uma estrutura, e um formato a seguir. E outros objetos, como esse ali (gravetos de madeira) não, você vai ter que criar e você vai ter que fazer. Então ele vai ter contato com tudo isso. Eles vão começar a perceber né. E nessa questão das regras, de jogos, dessas atividades que tem objetivo, eles vão aos poucos entendendo. Tem coisas que posso fazer assim, tenho a liberdade de criar, e tem coisas que não, tenho que pensar, tenho que seguir certas regrinhas pra chegar no objetivo. Até conseguir alcançar.

[Pesq] Eu estou fazendo um trabalho voltado pra crianças, mas não tenho muito contato com criança né. Tenho um sobrinho de 8 anos...

[Prof_3] É porque quem tá na academia, quem lida com as outras áreas, mesmo professor "de área né", ele não vai ter essa didática que a gente tem diferente. Ele vai tentar ver o desenvolvimento infantil, como que eles vão aprender, de que forma... Mas isso é interessante porque vocês também contribuem conosco né, porque a gente não tem essa visão que vocês tem né. De elaboração, de estruturação de pensamento. Assim, criar este tipo de jogo. Então é uma coisa muito legal. Então a gente vai ver se isso tem coerência ou não né. Aplicando com as crianças. Alguma coisa a gente pode achar que vai dar certo e as vezes não gosta. Mesma coisa a gente. As vezes a gente planeja e eles não gostam. Então não foi "aquilo tudo". Há momentos em que acontece isso também com a gente. A mesma coisa é o desenvolvimento de um jogo, depois tu vai testar pra ver se realmente pega esse

público, se este público vai conseguir, pra poder aprimorar né, é bem isso mesmo.

[Pesq] Aqui eu cheguei, na outra sala, e funcionou tudo bem. Aqui já tive bem mais dificuldade.

[Prof_3] Essa sala aqui eu não sei, se é porque ela fica bem no meio e pega bem a banda das duas, e sempre não consegue (conectar no wi-fi). É bem difícil. Até o datashow também é.

[Pesq] Vocês usam? (datashow).

[Prof_3] Sim, mas agora ele está com uns probleminhas técnicos. O mouse né, ele não tem o plug que você pode comprar. Você vai ter que levar na assistência técnica pra poder soldar porque ele é interno. O fiozinho é interno. Não dá pra gente usar.

[Pesq] Talvez usar televisão.. tem TV aqui..

[Prof_3] É, essa é antiguiinha, mas a mais moderna fica ali fora.

[Pesq] Nessa ideia de projetar as coisas no chão e ter uma interação

[Prof_3] isso é legal, isso é muito bom, gostei bastante. Eu achei que você ia colocar ali, como a gente tá acostumado né. Mas no chão foi muito legal...

[Pesq] E tu imaginas alguma atividade, que você falou né sobre não ser estruturado, onde as crianças pudessem inventar alguma atividade que usasse essa ideia de projeção, e de ah, perceber que uma coisa chegou perto da outra e fazer um som. Tu imagina alguma atividade que desse pra fazer, usando isso?

[Prof_3] Usando esse... essa tecnologia de projetar e fazer uma brincadeira.

[Pesq] Isso.

[Prof_3] Ah, legal... Poderia ser... tem que pensar né... Poderia até fazer esse jogos de tabuleiros, jogos de tabuleiro. Dependendo onde você vai tem que avançar na casa, retroceder, voltar no jogo... que é interessante também né pros menores, porque eles gostam bastante. Tem jogos mais simples, jogos mais complexos. Tipo um caça palavras ali, pras crianças que são alfabetizadas seria interessante. E daí eles vão descobrir uma letra, uma palavra, ou a figura e ter que descobrir... igual eu fiz com eles ontem, tipo uma caça ao tesouro. A gente escondeu os dinossauros e aí eu gravei num aplicativo o monstrinho falando que escondeu e tinha pistas. Aí tinha várias perguntas. Então pode desenvolver um jogo que tenha essas perguntas e eles vão ter que adivinhar onde está e voltar no jogo. Então você faz

um jogo aqui parado e um jogo iterativo procurando coisas. "Em certo local", e dá as características do local pra criança pensar, qual é esse local que tem essas características? Ah, tá lá, lá está a pista. Pega a pista e volta no tabuleiro. Faz uma jogada de várias coisas. Aí tem mais outra pista, em outro local. Vai ter que decifrar um enigma... então na verdade dá pra usar essa mesma tecnologia nesse sentido né. Ou montar as peças do próprio robô, pros mais avançados. Cada local tem uma parte da peça que vai dar início pra continuar o jogo.

[Pesq] Algo que não fica só parado.

[Prof_3] Não, fica dinâmico. Você vai ter que movimentar a escola toda. E pode outras turmas participarem, de vários enigmas, de várias coisas. Pode ser a escola inteira no mesmo jogo. Então cada sala vai ter uma tarefa diferente. E no final soma os pontos pra ver quem foi a turma que teve mais pontos que acertou. E pode ter vários temas. Por exemplo. O nosso projeto de instituição é sobre música. É ligado à música na Educação Infantil, e são várias possibilidades. Esse é o nosso tema. Então a gente faz atividades voltado a isso. Então se a escola tem esse tipo de jogo iterativo, que possa incluir, cada escola possa agregar, pode desenvolver e movimentar a escola toda naquela jogo ali. Um único jogo, várias participações. Pra vocês que vão pensar na tecnologia a ser desenvolvida...

[Pesq] Pois é, muitas vezes a gente sabe como fazer, mas "o que" fazer é o mais difícil. Faltam ideias.

[Prof_3] Mas tem vezes que parece que o cérebro secou, não tem mais nada. Eu acho que assim são os escritores né. As vezes eu tô inspirada, como eu vi agora um outro jogo bem legal que a gente vai fazer também, de perguntas e respostas num boliche. A gente vai usar não um boliche, mas outros tubos. Foi uma ideia que eu vi e pensei que a gente pode adaptar pro que a gente tá estudando sobre os dinossauros e aí a gente vai poder perguntar e saber um pouco deles. Cada vez que eles vão derrubando, cai e a pergunta é colocada pras equipes. Foi uma inspiração. Ainda bem que encontrei isso, foi um outro jogo, de outra coisa, mas que me inspirou a fazer. Assim a gente vai, a gente se inspira em alguma coisa e "ah, ainda bem que veio isso". Senão a gente fica sem ideias.

[Pesq] Um teórico que a gente estuda, o pai da informática na educação, o Seymour Papert, fala sobre as ideias poderosas. Tem toda um sequencia sobre o que é uma ideia poderosa. Por exemplo, a escrita. É algo que te ajuda a pensar. Não tá ligada a uma cultura específica, serve pra todas as culturas. Serve pra construir coisas, inventar coisas. São blocos de palavras que vai montando ao infinito de textos, de livros. Tem mais uns critérios... O LEGO por exemplo, permite criar coisas infinitas. E da mesma forma esta ideia que tu falou, é uma atividade que podes utilizar pra qualquer tema, qualquer

pergunta. A programação. Assim como a escrita, tu consegue criar infinito algoritmos.

[Prof_3] Isso aí acho tão... a gente que é leiga, olha assim, nossa, aquelas várias sequencias, aquelas coisas... Fica assim "nossa".

[Pesq] A gente tem essa ideia de que é difícil... Eu entrei num curso de informática por acaso. Quando eu ia jogar no computador, desligava o computador porque não sabia fechar a tela. As vezes não é tão complicado. Querendo ou não, o que a gente fez hoje foram algoritmos.

[Prof_3] Olha, interessante.

[Pesq] É uma sequência de passos, pra chegar num objetivo. A nossa vida são muitos algoritmos. Ah, vou botar uma roupa. A gente segue um algoritmo que é uma sequência de passos pra gente conseguir resolver um problema. Você pensando na tua aula, tem uma sequencia de passos pra resolver. As vezes, a gente pensar um pouco melhor, ah, posso fazer isso antes, ou isso depois, ou duas coisas ao mesmo tempo. Minha mãe faz isso sem ninguém ter ensinado, ela estudou até a terceira série. Ela bota uma coisa pra começar, e vai fazer outra porque consegue fazer mais de uma coisa ao mesmo tempo. Isso são algoritmos, a única diferença é a forma como a gente escreve, que não é no computador.

[Pesq] Alguma ideia que passaram na minha cabeça, sobre aplicabilidade deste projeto. Gostaria de ver se achas que tem lógica ou não. Outros blocos que daria pra colocar, pro brinquedo rope poderia agir diferente. Um deles eu pensei, seria um bloco de caneta. A criança colocaria o bloco, e a partir daquele bloco, onde ele andasse iria fazer um risco. Seria o baixar a caneta. Como se tivesse uma caneta dentro dele. E teria outro bloco de erguer a caneta, que a partir dali ele iria parar de desenhar. Seria algo sem um objetivo dado, a criança poderia desenhar o que quisesse.

[Prof_3] Ela é que vai riscar...

[Pesq] Teria um bloquinho de caneta. Teria um desenho que seria uma caneta abaixada. A partir daqui ele baixaria a caneta, mas ele não tem uma caneta de verdade.

[Prof_3] Ele vai marcar, mas virtualmente, isso? Aí vai ficar o rastro do trajeto que ele fez? Ah, interessante também.

[Pesq] Isso serviria pra desenhar qualquer coisa. Poderia ter um bloco pra dizer quando ele vai girar, em vez de girar 90 graus. Dizer, ah, ele vai girar mais que 90, então ele em vez de desenhar um quadrado, ele vai desenhar um triângulo por exemplo. Outros blocos que imagino. Poderia ter blocos de som, talvez. Ele chegou ali, ele faz um som, que a criança pode gravar talvez.

[Prof_3] Ou ele fez um som e aparece... vamos supor, se for um instrumento musical, se aparece um som daquilo ele pode estar identificando, por exemplo, deu um som de trovão, tem a figura do trovão e ele vai ter que associar. É um outro jogo no mesmo (estilo), mas com outra função, que eu posso associar. Porque quem tá fazendo associação de sons, né, discriminação auditiva, brincadeiras de discriminação auditiva, ele pode chegar em determinado local, fazer um som, e daquilo a criança vai ter identificar qual é pra seguir adiante. É um outro objetivo dentro desse que tu falou. Ou até uma letra. Ele chega até uma letra, ele faz o som. Ou até um número. Pode ter vários tipos. Perguntas e respostas. Você pode usar pra vários fins. Então você pode, ter várias funções, pra ajudar o professor na sala. Como você falou, em vez de eu sair pra ir na sala de informática, vem pra auxiliar naquilo que o professor tá dando. Porque as vezes é difícil você fazer a multidisciplinaridade que falam. Mesclar os temas. Porque os conceitos são juntos, a gente que separa para fins didáticos. E você fica naquilo. Uma coisa não tem conexão com a outra. E aí num jogo assim ele vai ter a conexão de todas, de ciência com matemática... As coisas não são dissociadas, a gente dissocia e o cérebro fica todo embananado, são coisa juntas né. Então essa seria uma questão que vocês poderiam estudar neste sentido, de ajudar o professor a fixar os conteúdos com essas perguntinhas, que essa coisa que o robozinho vai... Porque não é só comer a maçã, ele tem que comer a maçã e cumprir certas coisas. E pode associar, além da brincadeira, ele vai ter conhecimento. Algo que não é massante, ele vai brincar, que vai ser legal, que vai descobrir. Porque criança gosta muito de enigmas. Associar isso a enigmas pra descobrir, pra ir... Ele vai fazer a marquinha da caneta, e com isso ele vai mudando de sequência, pode mudar de cor.

[Pesq] Obrigado pela ajuda, vai ser muito útil...

[Prof_3] Ah, que bom. Então certo. Se precisar de alguma coisa... [fim

Dia 3

[Pesq] Então, fiz umas perguntas, pois eu falei né, que pesquisa é qualitativa. Eu preciso ter uma noção geral de como é, não apenas este trabalho que desenvolvi. Então eu queria que você falasse um pouco sobre o teu trabalho, de quanto tempo trabalha com crianças, e como é a rotina.

[Prof_4] Deixa eu te falar. A gente aqui trabalha com uma professora e com uma auxiliar. Então a professora gerente da sala é ela, eu sou só a auxiliar. Apesar de eu ser formada e pós graduada em pedagogia. Tu não tem problema disso?

[Pesq] Não.. Eu não sabia qual era a professora, e como falei contigo primeiro...

[Prof_4] Porque ela não gosta, não entende muito de tecnologia, ela disse "fica pra ti"... porque

a gente trabalha bem em conjunto assim, eu e ela.

[Pesq] Legal, também posso perguntar algo pra ela então depois..

[Prof_4] É, não sei se ela quer responder...

[Prof_5] Pode ser...

[Prof_4] Disse pra ele, que a gente trabalha bem em conjunto.

[Prof_5] É, a gente trabalha juntas, então o que ela falar tá perfeito. Tá bom pode ficar a vontade.

[Pesq] Então, é sobre isso né, como é a rotina, de como é o trabalho com as crianças..

[Prof_4] Neste ano tu quer saber?

[Pesq] No geral, desde quanto tempo trabalha com crianças, qual é tua experiência...

[Prof_4] Eu já to trabalhando com Educação Infantil já faz uns 15 anos. Vim morar pra cá, já comecei, fiz a faculdade de pedagogia, sou pós graduada em gestão, orientação e supervisão. Aí fiquei sete anos afastada da sala, fiquei na direção, fiquei como diretora da creche do Bairro_1, não se se tu conhece.

[Pesq] Estudei no IFSC, mas não conheço.

[Prof_4] É bem em frente. Tem a escola e tem a creche. Eu fiquei sete anos de secretária, depois fiquei quatro de diretora, daí eu vim, fiquei de diretora da creche ali da Marinha, ali onde chamam de Marinha. Aí eu cansei, daí tava com saudade de ficar na sala, e como moro aqui pertinho vim pra cá. Voltei pra sala em 2017. Que é o que eu gosto, estar com eles, esse retorno deles que eles te deram hoje assim, eu sou apaixonada. Por mais que eles não saibam, mas como eu disse, eles acabam dando umas dicas, eles soltam umas coisas que a gente nem imagina né. As vezes a gente faz um planejamento e sai exatamente outro porque eles vem com ideia que a gente nem pensou que a gente nem imaginava que eles iam perceber, então acabam mudando assim, mas isso é o que me faz me apaixonar pela Educação Infantil assim, essa falta de maldade que eles tem né, porque eles fazem as coisas por pura inocência. Então a nossa rotina agora com a pandemia mudou um pouquinho, então a gente tá com a turma dividida. Essas crianças vem essa semana, depois ficam em casa e vem o outro grupo. Então é uma experiência nova também que a gente tá vivenciando né, devido à pandemia. Eu começo aqui

no CDI às 8, aí eles tem o café da manhã, depois tem o planejamento. Vinte pras onze é o almoço deles, eles almoçam escovam o dente e começam ir embora. À tarde vem uma outra turma, daqui a pouquinho eles vão pro café, às 15 horas. E também né, nos outros dias que tu não tá aqui a gente faz o planejamento, faz outras coisas, faz bastante brincadeira. Essa idade que eles tão vai de 4 a 5 anos. Tem os que fizeram 4 anos até 31 de março, agora a partir de abril a gente tem aqueles que vão começar fazer 5 anos. Aí a gente já iniciou bastante assim, a questão de números né, mas tudo brincando né. Esse trabalho aqui eles que foram no computador, eles pintaram, eles foram colocando as quantidades, a letra dos nomes, a gente começou com a inicial. Então é uma amostra pra eles de como que é a diferenciação de números, de letras, mas nada alfabetizando, porque eles vão ter tempo bastante pra isso. Então é tudo em cima de brincadeiras, de jogos, como tu trouxe pra gente, foi bem legal. Como te disse, o olhar pedagógico ali são cores, são números ali, como eles contaram quantas vezes vai pra frente, quantas vezes vai pra trás, achei bem bacana assim, bem legal. Eles terem que pensar se vai passar da maçã, vai ter que voltar, vai ter que virar, vai ter que... acho que ali trabalha muita coisa junto, muita coisa num brinquedo só, achei bem bacana assim. Então é o que a gente trabalha aqui, é a brincadeira, o quebra cabeça... é montar pecinhas, é eles contarem o que vai precisar pra montar, o que vai precisar ter.

[Pesq] Eu percebi que tinha uns quebra-cabeças sobre a mesa, tive a impressão de que conseguiram encaixar bem as pecinhas (do projeto), apesar de não ser um formato bem certinho como no quebra-cabeça... Eles foram encaixando naturalmente.

[Prof_4] É como eu te disse hoje de manha. Devido à pandemia, como eles ficaram em casa um ano, a gente percebeu que esse ano tá bem diferenciado. Quando eles passavam de uma turma pra outra, iam mais ou menos igual, todos sabendo a primeira letra do nome.. todos sabendo os números de 0 a 9 né, e esse ano não, esse a gente percebeu bastante diferença por eles terem ficado esse ano em casa. Então veio, como hoje de manhã a Menina_0, a Menina_1, que já pegaram né, a Menina_2, já tão sabendo os números, já sabem as letras, então são crianças que talvez os pais ficaram mais em cima em casa, fizeram as atividades online... E aí já tem esses, como tu viu a tarde, que já tiveram um pouquinho mais de dificuldade né, de raciocínio, de manter a questão de quantidade, esse ano de pandemia a gente notou bastante essa diferença.

[Pesq] Tem um impacto grande na criança.

[Prof_4] Isso. Também é entendível né, porque muitos pais saíam pra trabalhar, e as crianças ficavam com vó, não conseguiam acompanhar as atividades online que a gente mandava pra casa... e aí faz falta né essa convivência, porque acaba um ensinando o outro. "Ah, esse aqui é 2 porque tem duas

coisas", mesmo que eu não digo mas tem um que já se mete, já diz e em casa não tem isso né. Então a gente esse ano tem essa missão de igualar um pouquinho eles, pra ano que vem que eles vão pro pré, que ainda é aqui, e depois eles vão pra escola.

[Pesq] O que você pensa sobre as crianças desta idade (4 a 5) tendo contato com tecnologia.

[Prof_4] Ah, eu acho necessário. Eu acho necessário mas também tem que ter o equilíbrio assim. Tu não pode deixar a criança só com tecnologia, vamos supor, o celular, que é o que eles tem em mãos, né, que a gente viu que foi o que aconteceu com alguns, que passaram a pandemia toda com o celular na mão, tem que ter o equilíbrio, mas também acho que com a pandemia a tecnologia veio mais ainda né, e a gente viu até com os professores. Como eu te disse, a Prof_5 já é uma pessoa um pouquinho mais velha, então ela tem grandes dificuldades com a tecnologia. A gente até recebeu da prefeitura uns chrome books, então eu estou dando bastante, apesar de eu não ter esse conhecimento todo, porque eu já me acho ultrapassada né, mas aí ela eu tô ajudando bastante ela nessa questão, porque o chrome book é um pouquinho diferente. Mas eu disse pra ela, depois que aprender a mexer ele é muito prático, porque fica tudo gravado no drive, do drive já consigo mandar imprimir, então achei ele bem prático. Mas de todas as professoras aqui, eu acho que só nós estamos usando o chrome book, porque as outras tem essa dificuldade. Então eu penso que se as crianças que estão nascendo, que estão nessa idade, não tiverem contato com a tecnologia, vão chegar umas pessoas mais velhas igual nós assim né, meio que analfabetos na tecnologia, e eu acho que a pandemia veio para a tecnologia ainda mais pra nossas vidas, porque tudo hoje em dia né.. A gente mesmo teve que se adaptar à aulas online, à internet, à plataforma, e aí que vai né. Eu acho que eles precisam estar por dentro disso tudo senão ficam pra trás no mercado de trabalho também.

[Pesq] Uma coisa que defendo neste trabalho, na questão dos bloquinhos, é a visibilidade. A criança poder ver os blocos, e poder ver o programa que ela tá criando. Diferente de tá inserindo o programa nos botões, e não ver esse programa criado. Uma outra criança que tá ali do lado ela também pode ver esse programa. Eu queria saber tua visão a respeito de outras atividades que envolvem essa questão da visibilidade, onde a criança tem contato com algo que ela está vendo na frente dela, o que isso impacta nas outras atividade.

[Prof_4] Eu acho que isso entra no equilíbrio também, porque aí eles não ficam só na tecnologia, mas eles precisam vivenciar e visualizar com os bloquinhos, então não fica só no apertar o botão, como no celular, pode ver que eles apertar, apertar ali né. Então se tu não chamar a atenção ali pros bloquinhos... Então acho que é o equilíbrio ali da tecnologia, não fica só no apertar o botão, mas sim em terem que olhar os bloquinhos, ter que visualizar, ter que fazer a sequencia, pra tecnologia poder

andar né. Acho que faz eles pensarem, faz terem que prestar atenção nos bloquinhos, e dá o equilíbrio da tecnologia.

[Pesq] Eu vejo essa questão por exemplo aqui nesse painel. Tem os números né. Tá uma coisa bem exposta, a criança chega e tá vendo isso na frente dela, diferente de falar "um", o que é um?

[Prof_4] E até foi engraçado ontem, isso aqui foi uma proposta de três dias. Primeiro eles pintaram o cartaz, que a ideia era eles escreverem os números, alguns conseguiram definir um pouquinho mais, alguns não, e aí depois eu chamei, foram só três lá no computador da secretaria, pra imprimir os números né. Eu disse "Vamos com a Prof_4 lá", e eles "fazer o que?" "Imprimir os números no computador, vai sair numa impressora", e eles "você vai deixar nós mexer no computador?" Parece uma coisa que vocês não podem tocar. "Sim, a gente vai mexer", mostrei onde ficavam os números. Então acho que é interessante eles escreverem, pintar com lápis, com caneta, como foi no nosso tempo, mas é interessante também eles irem lá e ver que os números saem de uma máquina também né.

[Pesq] A gente olha e não percebe os detalhes... Tem um número desenhado ali..

[Prof_4] Como eu disse, eu as vezes não vou perceber no teu trabalho tecnológico algumas coisas como você não vai perceber no trabalho pedagógico outras.

[Pesq] Nessa ideia de projetar alguma coisa no chão, e ter essa interação de um robô, onde se tem a percepção de algo que chegou perto de algo, e aí acontece um evento. No caso ali é um robô que chega perto da maçã, e aí faz o som de comer a maçã. Tu imagina, alguma atividade que pudesse usar esse mesmo princípio, desta mecânica, vamos dizer assim? Uma imagem, um cenário no chão, e algumas coisas espalhadas assim que pudessem interagir.

[Prof_4] É, tipo se fosse uma contação de história? Vai contando uma história e o robô vai demonstrando um cenário, vai chegando perto do cenário e vai fazendo um personagem da história. Eu pensaria nisso... Tipo um lobo, chegou no mato e o robô segue o lobo, algo assim, e vai chegando...

[Pesq] Por exemplo, se chegou perto do lobo, faz o barulho do lobo...

[Pesq] Que problemas que tu imagina que este tipo de projeto pode ter no dia a dia, aqui, vamos dizer, de uma creche, no teu trabalho, que sugestões poderia dar pra melhorar o projeto. Pra melhorar em algum aspecto.

[Prof_4] Eu não vi problema nenhum... na minha área pedagógica... na tua área as vezes tu pode ter percebido algumas mudanças em algumas coisas, mas assim...

[Pesq] Uma apreensão que eu tinha é, pela estrutura a criança derrubar e quebrar o projetor.

[Prof_4] Mas aí, vamos supor que tu deixasse esse equipamento na minha mão. Antes de tudo isso eu iria explicar pra eles que isso ali é um projetor, que teriam que cuidar pra não bater, porque também a gente trabalha com projetor na parede né, então quando a gente traz a gente diz que só a prof pode mexer, eles não podem mexer. A gente tá usando o chrome book agora, então eles também tem esse contato assim. Eu não vejo como um perigo pra eles. Isso não ia cair em cima de uma criança e machucar. No máximo iria quebrar, no máximo teria um prejuízo. A gente se preocupa mais se fosse uma coisa muito pesada e poderia machucar uma criança. A gente iria trabalhar com eles o cuidado com o equipamento. Tu viu que foi o que gerou bastante curiosidade pra eles né. Porque quando eles chegaram aqui, eles questionaram o que era. E a outra professora disse que tu iria mostrar um robô e imaginaram que isso ali fosse o robô. Por isso ficaram com toda essa expectativa, porque imaginam que o robô grande...

[Pesq] Até na sala da Prof_3 eles fizeram um robô lá...

[Prof_4] Eu acho que isso aqui, daqui pra frente, vai render outros planejamentos, porque daí vão começar a perguntar, disso, vão começar lembrar.. Os pais vão vir nos questionar, porque daí eles chegam em casa e começam a conversar. E daí da fala deles, do questionamento, a gente dá uma continuidade no planejamento.

[Pesq] Acho que, nossa, vocês me ajudaram de mais assim. Existe a parte de programação, e etc. Mas isso não vai ser nem de longe o forte do trabalho, vai ser o que eu peguei aqui, o que aconteceu. Vão me perguntar como as crianças reagiram.

[Prof_4] Mas eles são muito imprevisíveis, não são?

[Pesq] Pois é, e as vezes a resposta que parece óbvia falam algo totalmente diferente.

[Prof_4] Por isso estava sentido contato assim, na direção você trabalha mais com os adultos, a prefeitura, os pais, os funcionários, então chegou num ponto que o adulto me encheu o saco assim, aí quando voltei pra sala assim tem muitas vezes que tenho que sair, virar e rir, porque não pode dar aquela confiança todas, aí a gente ri assim, depois volta... Mas é bem apaixonante..

[Pesq] Agradeço a ajuda.

[Prof_4] A gente achou bem bacana também. A gente abre as portas aí. A professora liberando, se tu precisar voltar.. estamos aí. [fim]